AFFDL-TR-67-47

THE METHOD OF CHARACTERISTICS FOR THREE-DIMENSIONAL REAL-GAS FLOWS

INCLUDING THE DIGITAL COMPUTER PROGRAM DETAILS AND OPERATING PROCEDURES FOR FROZEN FLOWS

Charles R. Strom
Cornell Aeronautical Laboratory, Inc.
Buffalo, New York 14221

Technical Report AFFDL-TR-67-47

July 1967

Distribution Of This Document Is Unlimited

AIR FORCE FLIGHT DYNAMICS LABORATORY
RESEARCH AND TECHNOLOGY DIVISION
AIR FORCE SYSTEMS COMMAND
WRIGHT-PATTERSON AIR FORCE BASE, OHIO

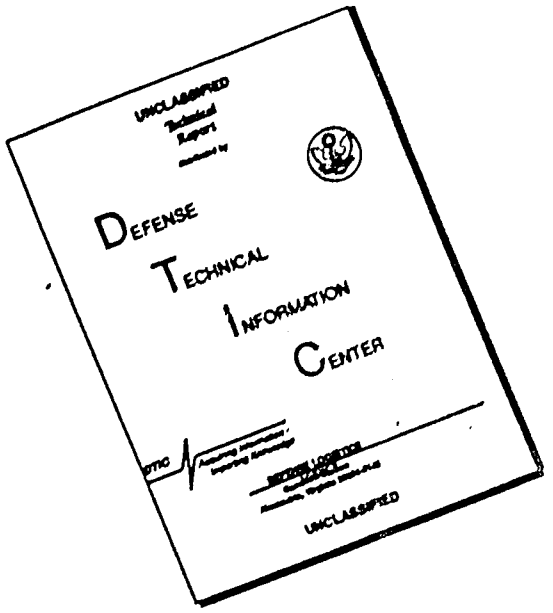
241

NOTICES

When Government drawings, specifications, or other data are used for any purpose other than in connection with a definitely related Government procurement operation, the United States Government thereby incurs no responsibility nor any obligation whatsoever; and the fact that the Government may have formulated, furnished, or in any way supplied the said drawings, specifications, or other data, is not to be regarded by implication or otherwise as in any manner licensing the holder or any other person or corporation, or conveying any rights or permission to manufacture, use, or sell any patented invention that may in any way be related thereto.

Copies of this report should not be returned to the Research and Technology Division unless return is required by security considerations, contractual obligations, or notice on a specific document.

DISCLAIMER NOTICE



THIS DOCUMENT IS BEST QUALITY AVAILABLE. THE COPY FURNISHED TO DTIC CONTAINED A SIGNIFICANT NUMBER OF PAGES WHICH DO NOT REPRODUCE LEGIBLY.

AFFDL-TR-67-47

THE METHOD OF CHARACTERISTICS FOR THREE-DIMENSIONAL REAL-GAS FLOWS

INCLUDING THE DIGITAL COMPUTER PROGRAM DETAILS AND OPERATING PROCEDURES FOR FROZEN FLOWS

Charles R. Strom

Cornell Aeronautical Laboratory, Inc.
Buffalo, New York 14221

FOREWORD

This final report was prepared under Contract Number AF33(657) - 10423 by the Cornell Aeronautical Laboratory, Inc. The contract was initiated under Project No. 1366, "Aerodynamics and Flight Mechanics," Task No. 136615, "Experimental and Theoretical Investigation of Aerodynamic Heating of Super-Orbital Speeds." The research was administered by the Air Force Flight Dynamics Laboratory, Research and Technology Division, Air Force Systems Command, Wright-Patterson Air Force Base, Ohio, with Mr. R.D. Neumann of AFFDL (FDMG) as project engineer. The final version of the program was developed between November 1965 and December 1966. The author released the manuscript for publication in January 1967.

The author is indebted to Dr. A.F. Burke and to Dr. J.T. Curtis for advice and assistance during this investigation.

This report is also being published as a Cornell Aeronautical Laboratory Report No. AM-1800-Y-4.

This technical report has been reviewed and approved.

Philip P. Antonatos

PHILIP P. ANTONATOS
Chief, Flight Mechanics Division
Air Force Flight Dynamics Laboratory

ABSTRACT

The method of characteristics is formulated for the computation of the supersonic flow of an inviscid, reacting gas over a smooth three-dimensional body. Various methods of constructing networks of bicharacteristic lines are examined from the point of view of numerical stability and accuracy. A new method of forming the network, which consists of projecting forward along streamlines from data points on specified data planes, is found to be most easily adopted to the particular requirements of nonequilibrium chemistry.

The general method was coded for the IBM 7090 computer and the program demonstrated for the case of an ideal gas. Calculations were made for the flow about a spherical-tip 15° half-angle cone at 10° angle of attack and a generalized elliptical body at zero incidence. Since the program yields the pressure distribution along specified streamlines, it is straightforward, in principle, to link it to a finite-rate chemistry stream tube program to treat three-dimensional, nonequilibrium flows.

TABLE OF CONTENTS

TABLE OF CONTENTS (Cont.)

LIST OF ILLUSTRATIONS

LIST OF ILLUSTRATIONS (Cont.)

LIST OF ILLUSTRATIONS (Cont.)

LIST OF TABLES

SYMBOLS AND NOMENCLATURE

$A, B, C,$ etc.	- Coefficients defined as used.
$A1_i, B1_i, C1_i,$ etc.	- Coefficients in polynomial surface fits for function i.
$A_{\mu\nu}(\phi)$	- Functional form defined by(Equation(9))
a	- Frozen speed of sound. (Equation(34))
a_{ik}	- Matrix of coefficients of quadratic forms defining normal cone in steady and unsteady flow.
$a_{\mu\nu_i}$	- Coefficient matrix for equations of motion.
$a'_{\mu\nu_i}$	- Transformed coefficient matrix (Equation(6))
$B(x, y, z)$	- Body function for steady flow.
B_μ	- Vector representing right hand side of transformed equations.
b_μ	- Given functions of u_ν and x_i.
c^2	- $q^2a^2/(q^2 - a^2)$
c_p	- Specific heat at constant pressure. (Equation(28))
c_v	- Specific heat at constant volume (ergs/mol deg. K)
dS	- Defined in(Equation (77))
E	- Number of elements in chemical system.
$e^{(n)}$	- Error between pressures determined from shock relations and compatibility equations for n^{th} iteration cycle.
$\tilde{f}$	- Finite rate term in compatibility(Equation (33))
$\hat{f}$	- Function defining characteristic.
$\hat{g}, \hat{h}, \hat{k}$	- Arbitrary functions forming an independent set with $\tilde{f}$.
$(G_i)_j$	- See(Equation(73))
h'_i	- Enthalpy of i^{th} species (ergs/mole)
h'	- Static enthalpy = $\sum_{i=1}^{NS} \gamma'_i h'_i$
h_i	- Enthalpy of i^{th} species $h_i = \frac{h'_i}{RT'_\infty}(MW'_\infty)$
H_∞	- Dimensionless total enthalpy (Equation 127)
h_i°	- Heat of formation of i^{th} species.
h_μ	- Set of linear factors ($\mu = 1, 2, 3, 4$).
I_i	- Enthalpy of i^{th} species including energy of translation and rotation.
$\bar{i}, \bar{j}, \bar{k},$	- Unit vectors along $(x, y, z,)$ axes.
K	- Pressure gradient = $(p_2 - p_1)/p_1$

k_{b_i}, k_{f_i} - Backward and forward rate constant for i^{th} reaction resp.

$\bar{K}_1$ - $\partial \bar{N}_s / \partial x$

L' - Length used in nondimensionalizing

$\bar{L}$ - Characteristic length.

$\bar{L}, \bar{M}, \bar{N}$ - Characteristic coordinate system.

M - Mach number.

M_j - Chemical symbol for j^{th} species.

$MW = \left(\sum_{i=1}^{NS} \gamma_i\right)^{-1}$ = molecular weight of mixture

$MW'_\infty = 1 / \sum_{i=1}^{NS} \gamma'_{i\infty}$ = molecular weight of free stream

$\bar{N}_S$ - Normal vector to shock surface.

$\bar{N}_B$ - Normal vector to body.

NR - Number of Reactions

NS - Number of species.

NV - Number of species not in vibrational equilibrium.

p - Pressure = $p'/\rho'_\infty u'^2_\infty$

$\bar{Q}$ - Generalized velocity vector.

Q - $|\bar{Q}| = (u^2 + v^2 + w^2 + 1)^{1/2}$

Q_i - Given concentration source term for species i.

$\bar{q}$ - Velocity vector

q - $|\bar{q}|$

R - Total number of reactions considered.

$\mathcal{R}'$ - Universal gas constant = 8.313405×10^7 ergs/mol °K

$R' = \frac{\mathcal{R}'}{MW'}$ - Gas constant

$\bar{\tau}$ - Vector in direction of velocity component tangent to shock surface.

T - Temperature

t - Time

u'_∞ - Free stream velocity

u - Velocity component along x-axis.

u_ν - Each of several velocity components ($\nu = 1, 2, 3, \ldots$).

v - Velocity component along y-axis.

w - Velocity component along z-axis.

x, y, z - Cartesian coordinate system.

x_i - Each of a set of independent coordinates ($i = 1, 2, \ldots$).

x_i' - Each of a set of independent coordinates ($i = 1, 2, \ldots$).

α - Angle-of-attack.

α, β_i - Arbitrary set of orthonormal vectors ($i = 1, 2, \ldots$).

α_{ik} - Atoms of element k in i^{th} species.

β - Mach angle

Γ - Ratio of specific heats.

γ_i - $\gamma_i' / \sum_{i=1}^{NS} \gamma_{i\infty}'$

γ_i' - Specific molar concentration of i^{th} species in moles/gm or equivalent units.

δ - Parametric angle for bicharacteristic defined in Figure 1

ϵ_j - Vibrational energy of j^{th} species.

θ - Flow angle for steady flow shown in Figure 1.

Λ - $\frac{U_\infty'^2 MW_\infty'}{RT_\infty'}$

$\Re$ - $1/\Lambda$

μ - Angle defined in Figure 19 = α

ν_{ij}, ν_{ij}^* - Stoichiometric coefficients

ξ - Coordinate along the streamline.

$\bar{\xi}, \bar{\eta}, \bar{\zeta}$ - Orthogonal unit vectors defining cartesian coordinate system defined in Figure 1.

ρ - Density

σ - Shock angle = angle between free stream velocity vector and normal to shock minus 90°.

$d\tau$ - Time taken for propagation of a wave along a ray corresponding to a bicharacteristic.

π - Flow angle defined in Figure 21.

ψ - Flow angle defined in Figure 1.

ω - Flow angle defined in Figure 21.

SUBSCRIPTS

$(\ )_i$	where (i) = 1, 2, etc. - refer to quantities at given points.
$(\ )_\infty$	evaluated in free stream
$(\ )_{x,y}$	x, y, etc. either a derivative or components along respective axis
$(\ )_o$	initial values
$(\ )_+$	evaluated upstream at shock point
$(\ )_-$	conditions across shock at shock point
$(\ )_T$	tangential conditions
$(\ )_N$	normal conditions
$[\ ]$	The brackets define jump conditions across the characteristic surface $x_i' = 0$, i.e. $\left[\frac{\partial f}{\partial x_i}\right] = \left(\frac{\partial f}{\partial x_i}\right)_+ - \left(\frac{\partial f}{\partial x_i}\right)_-$ where the + and - subscripts now apply at the characteristic surface rather than the shock.

DIMENSIONS

Primes refer to dimensional quantities except when used with x_i' coordinates. The normalizing relationships are listed below.

$$x = x'/L' \quad ; \quad y = y'/L' \quad ; \quad Z = Z'/L' \quad ; \quad t = t' \frac{U_\infty'}{L'} \quad ;$$

$$u = u'/u_\infty' \; ; \quad v = v'/u_\infty' ; \quad w = w'/u_\infty' ; \quad a = a'/u_\infty' \; ;$$

$$p = p'/\rho_\infty' u_\infty'^2 ; \quad \rho = \rho'/\rho_\infty' \; ; \quad T = T'/T_\infty'$$

$$MW = 1 \Big/ \sum_{i=1}^{NS} \gamma_i \; ; \quad \gamma_i = \gamma_i' \Big/ \sum_{i=1}^{NS} \gamma_{i\infty}' = \gamma_i' \; MW_\infty'$$

$$MW = MW' / MW_\infty'$$

$$h = \sum_{i=1}^{NS} \gamma_i \, h_i \; ; \quad h_i = h_i' \; MW_\infty' / \mathcal{R}' T_\infty'$$

$$h = MW_\infty' \; h' / \mathcal{R}' T_\infty' \; ; \; R_{Vj} = \frac{\partial \epsilon_j}{\partial t}$$

$$\Lambda = MW_\infty' \; u_\infty'^2 / \mathcal{R}' T_\infty' \; ; \; \mathcal{R} = \frac{1}{\Lambda}$$

$$Q_i = Q_i' \; L' \; MW_\infty' \; / \; \rho_\infty' \, u_\infty' \; ; \; \epsilon_i = \epsilon_i' \; MW_\infty' / \mathcal{R}' T_\infty'$$

$$I_i = I_i' \; MW_\infty' / \mathcal{R}' T_\infty' \; ; \; C_p = C_p' \; MW_\infty' / \mathcal{R}' ; \; C_v = C_v' \; MW_\infty' / \mathcal{R}'$$

$$\vec{q} = u \, \hat{i} + v \, \hat{j} + w \, \hat{k}$$

$$q = |\vec{q}| = \sqrt{u^2 + v^2 + w^2}$$

Section 1

INTRODUCTION

1.1 POSSIBLE METHODS OF SOLUTION

The general equations governing the three-dimensional, supersonic flow of a reacting gas are highly nonlinear and cannot be solved in closed form. Thus, without resorting to many simplifying approximations, the only alternative is to solve the equations numerically.

The prime objective of the present study was to develop a numerical procedure for determining the supersonic flow field about a general three-dimensional body including nonequilibrium chemical effects. Several techniques are available for doing this.

Most of these methods make use of the hyperbolicity of the general governing equations by formulating the problem as an initial value problem which can be continued in either time or space. One method, however, which has recently been proposed for steady flow by Telenin and Tinyakov[1] treats the problem as a boundary value problem which must be solved between two boundaries (i.e., the body and the shock) subject to certain known boundary conditions. This scheme, which has not yet been treated in detail in the literature, is somewhat similar to the direct method of Belotserkovsky for two-dimensional and axisymmetric flow fields in that polynomial approximations are used to reduce the partial differential equations to a set of ordinary differential equations which can be integrated numerically. Both subsonic and supersonic flows can be solved. However, since the method can probably not be easily extended to include complex chemistry models and unsteady flow problems, it will not be considered any further here.

Of the remaining possible methods only two appear to be capable of providing sufficient detail through the shock layer. The first of these, called the finite difference approach, consists of replacing the original partial differential equations by a system of difference equations - formed through direct substitution of finite differences for the derivates - which can then be solved using a step-by-step numerical scheme. In the second technique, generally termed the method of characteristics, the equations of motion are transformed to a characteristic coordinate system and the derivatives are again approximated by finite difference equations. The resulting equations are numerically integrated step-by-step along characteristic surfaces throughout the flow-field.

The finite difference techniques can be further subdivided according to the form in which governing partial differential equations are written and the differencing techniques employed. In general, the classifications would be as follows:

1. Standard (SFD) - If the equations of motion are written in either the Lagrangian or Eulerian form and the finite difference approximations are substituted directly, the scheme is termed the standard finite difference technique. Discontinuities are handled by imposing appropriate jump conditions (e.g., Rankine-Hugoniot conditions at shocks).

2. Finite difference technique utilizing artificial viscosity (PFD) - A small "pseudoviscosity" term is introduced into the nonviscous flow equations. The resultant effect is that the solutions remain stable even in regions of large gradients (e.g., near shock waves) Von Neumann and Richtmyer[2] obtained a solution to the one-dimensional, unsteady flow equations using this method which accounts for the presence of free boundaries (e.g., shock waves) automatically. Burstein[3] has recently extended this idea to two-dimensional, unsteady flows.

3. Finite difference procedure applied to conservation laws (CFD) - Lax[4] developed a method which treats the governing equations in conservation form and uses a differencing procedure which has the effect of introducing dissipative terms. The method can handle cases in which the solution has certain specific kinds of discontinuities (termed weak solutions). Lax and Wendroff[5], Burstein[3] and Bohachevsky have recently applied this method to various fluid flow problems. Also, Moretti[61] has used this procedure coupled with a method of characteristics approach at the boundary points to solve unsteady flow equations.

4. Finite difference procedure applied directly to Navier-Stokes equations (NSFD) - This procedure which is generally difficult because of the complexity of the differential equations involved was first applied by Ludford et al.[62], and more recently by Crocco[6] for one-dimensional unsteady flow.

While other methods have been proposed, these four appear to be the most useful for calculating multidimensional flow fields.

In Table I, a summary of some of the advantages and disadvantages of each of these finite difference approaches and the method of characteristics (MOC) approach are given.[7,8,9] On the basis of these comparisons the following conclusions appear warranted.

1. Although finite difference methods can treat shock waves and other discontinuities, their accuracy is impaired (especially near the shock) because they have the effect of smearing out the shock wave over several mesh points. The effect of this smearing on the rest of the flow field for multidimensional flows is not known at present.

2. Fixed boundaries are more accurately treated by the MOC technique when the body shape is arbitrary. The finite difference schemes appear to be better only when the body shape is such that points on the body lie along coordinate lines so that body points are mesh points. Otherwise, interpolations, which could lead to errors and/or possibly instability, are required.

3. The disadvantages of the method of characteristics approach are that the program logic required to code the procedure for the computer is quite complicated and that in order to obtain a solution at a given mesh point, considerable iterating is needed. This latter situation, of course, adversely affects the computing time required to obtain a solution.

TABLE I

COMPARISON OF POSSIBLE APPROACHES

		MOC	SFD	PFD	CFD	NSFD
1.	FREE BOUNOARIES (SHOCKS, CONTACT SURFACES, ETC.)	TREATED AS DISCONTIN-UITIES USING SPECIAL RELATIONS	TREATED AS DISCONTIN-UITIES USING SPECIAL RELATIONS	AUTOMATICALLY HANOLED - HOW-EVER RESOLUTION IS SPREAO OVER SEVERAL MESH WIDTHS	SAME AS PFO	SAME AS PFO
2.	FIXED BOUNDARIES	EXACT BOUNDARY IS SOLVED USING SPECIAL RELATION	SAME AS MOC	DIFFICULTY CAN ARISE BECAUSE ARTIFICIAL VISCOSITY IN-CREASES ORDER OF EQUATIONS THUS REQUIRING EXTRA BOUNDARY CONDITIONS	TREATED BY INTER-POLATION	TREATED BY SPECIAL RELATIONS
3.	NO. OF INDE-PENOENT VARIABLES WHICH CAN BE SOLVED	3-SPACE PLUS TIME	SAME AS MOC	SAME AS MOC BUT MORE OIFFICULT BECAUSE EXTRA BOUNDARY CONDI-TIONS MUST BE INTROOUCED	SAME AS MOC	SAME AS MOC
4.	STEP SIZES	POTENTIALLY LARGEST	SMALLER THAN MOC BECAUSE TRUNCATION ERRORS ARE LARGER	SMALL BECAUSE HIGH GRADIENTS REQUIRE A FINER MESH	SAME AS PFD	SAME AS PFD
5.	SOLUTION GRIO	UNEQUALLY SPACED	EQUALLY SPACED	SAME AS SFD	SAME AS SFD	SAME AS SFD
6.	ACCURACY	POTENTIALLY MOST ACCU-RATE BECAUSE THIS METHOD OF SOLUTION MOST CLOSELY FOLLOWS FLOW MODEL	INTERMEDIATE ACCURACY BECAUSE OF LARGER TRUNCATION ERRORS	LEAST ACCURATE BECAUSE ERRORS CAN BE INTRO-DUCED BY NOT PROPERLY SPECI-FYING EXTRA BOUNDARY CONDITIONS	ACCURACY IS IM-PAIREO BY SMEARING OUT SHOCK OVER SEVERAL MESH POINTS	SAME AS CFO
7.	PROGRAMMING DIFFICULTIES	BY FAR MOST DIFFICULT	OF INTER-MEDIATE DIFFICULTY	SIMPLEST	SAME AS PFD	SAME AS PFD

As noted initially, the prime objective of the present study was to develop a numerical procedure for determining the flow field about a general 3-D body including nonequilibrium chemical effects. Particular interest was centered in the afterbody or supersonic portion of the flow field because for a typical spherically blunt nose, the forebody or subsonic-transonic portion of the flow field can be obtained from a transformation of an appropriate axisymmetric solution except at high angles of attack. One naturally wonders whether this problem can be most efficiently treated using the method of characteristics or one of the finite-difference approaches. The answer to the question depends on three considerations:

(a) Computing time per solution

(b) accuracy and reliability of the solution near the shock and body

(c) prospects for treating chemical nonequilibrium effects.

When the present work was started in June 1963, there was not sufficient information available concerning the finite-difference methods to even begin to consider the answer to this question rationally. In fact at the present time, after 3-4 years experience with both general methods, the answer to this question is still not clear in the case of reacting gas flows. In 1963, it was decided to pursue the method of characteristics because considerable success had been attained using that method for 2-D and axisymmetric flows and it yielded results having the desired detail near the shock and body. It was also clearer how procedures for treating nonequilibrium chemical effects could be included in the program to various degrees of approximation (ex. chemically frozen along streamlines, finite-rate streamtube with locally frozen gas chemistry for obtaining flow quantities in the mesh calculation, etc.). Hence this report deals in detail with applying the method of characteristics to three-dimensional reacting flows and no further consideration is given to finite-difference methods. Questions regarding the relative merits of the two approaches must await further work.

1.2 SOLUTION OF HYPERBOLIC PARTIAL DIFFERENTIAL EQUATIONS USING THE METHOD OF CHARACTERISTICS

The theory of the method of characteristics as applied to quasi-linear, hyperbolic equations has been known since Monge and Ampere in the early nineteenth century. A monograph by Massau[8] first indicated the use of the theory to solve a system of two equations in two unknowns. Later, in 1928, Lewy[40] used the method to show that the initial value problem for a quasi-linear hyperbolic differential equation in two independent variables has a unique solution. Titt[11] later generalized the method of Lewy to the case of three independent variables. Since that time, the method has been utilized by several authors in formulating methods for solving many different hyperbolic problems in fluid mechanics. Ferri[12] and Meyer[13] discuss much of the earlier work as it applied to several different problems in fluid dynamics and give a complete list of references. Now, with the recent development of faster and larger electronic digital computers, more ambitious problems such as the one described here, may be attempted. Thus, it is now possible to consider the flow over general bodies exhibiting rather general motions and including complicated thermochemical effects.

The Cauchy problem (or initial value problem) considered here can be summarized simply as follows. Given an initial data surface and known boundary conditions (body surfaces or shock waves), the method of characteristics is to be used to obtain the solution of the governing partial differential equations at another surface separated in either time or space from the original one. The solution will be obtained at a discrete set of grid points on the new surface by integrating the governing equations numerically along certain selected characteristic lines. When as many new grid points as needed are obtained, the surface thus calculated is taken as the new initial value surface and the next adjacent surface is obtained similarly. The process is then continued step-by-step until the desired portion of the flow-field is solved. This technique has been used successfully by innumerable authors for solving flow fields in two independent variables. However, very little actual numerical work has been performed for problems involving either complicated thermochemical effects or more than two-independent variables.

It is the object of the present work to obtain a numerical method for solving just such problems.

Several authors have discussed the applications of the method of characteristics to multidimensional flow fields from a purely theoretical viewpoint without offering any numerical solutions. These include: (1) the pioneering efforts of Thornhill[14], who discussed two possible characteristic networks for three-independent variables, (2) Coburn and Dolph[10], who extended the work of Titt to fluid flow problems, (3) Clippinger and Giese[15], who used a generalized vector formulation to derive the basic characteristic equations and (4) Holt[16], who established a finite difference scheme based upon the earlier work of Coburn and Dolph. More recently, Fowell[17] has written the basic finite difference equations which apply to one of the networks which Thornhill proposed. Although, as will be discussed in Section 3. Fowell's method was shown to be unstable, it has proved to be very useful in developing stable multidimensional integration analogs.

A second group of authors have succeeded in obtaining some results using numerical computational procedures. The first few of these were limited to simpler calculations which were accomplished by hand with the aid of desk calculators. Among these are included the works of Ferrari[18], Moeckel[19] and Bruhn and Haack[20]. Later Butler[21] and Tsung[22] succeeded in obtaining some limited solutions by utilizing the earlier digital computers. More recently, Moretti et al.[23], Kackova and Cuskin[24] and Sauerwein[9] have published results obtained with the present generation of high speed computers. The most important of these various schemes will be discussed and compared in Section 3. It would appear that the possibilities in this area have just begun to be fully realized and with the addition of still faster and more efficient computers even more complicated problems should prove amenable to solution using the method of characteristics approach.

The extensions required to the MOC to include nonequilibrium chemistry effects have been set forth by Chu[25], Resler[26] and Wood and Kirkwood[27] among others. Sedney et al.[29] have applied the numerical method to calculate vibrational relaxation and Capiaux and Washington[30] and Eastman[31] have utilized a simplified reaction system without relaxation.

Zupnik et al.[32], and Widawsky[33] have recently developed and applied the theory for calculating nonequilibrium nozzle flows, while only Wood et al.[34] and Curtis[35] have applied the method to obtain numerical results for the afterbody flow fields including the full reacting and relaxing flow equations.

The present effort describes the development of a method which in principle is capable of calculating the flow field in the vicinity of a general three-dimensional body including nonequilibrium thermochemical effects. Since the development includes the unsteady flow equations, the scheme could eventually be used to calculate the flow properties in the vicinity of a blunt asymmetric body or a sharp cone at angle-of-attack. In Section 2 the basic theory of the method of characteristics as it applies to the present problem is given. The relations are written in terms of two special characteristic coordinate systems, each of which is thought to be best for numerical applications. The proposed integration network is presented in Section 3 and is discussed and compared with other schemes which have previously been utilized. The discussion includes the requirements which are thought necessary for insuring stability of a successful integration procedure. A description of the numerical scheme as it applies to each of three unit processes for calculating the flow properties at points in the field, on the body surface and on the shock wave surface is given in some detail, including the modifications necessary to account for nonequilibrium chemical effects.

The remainder of the work describes the actual machine program in which the methods proposed in Section 3 are applied to solve practical flow problems. In the present case, the program is limited to steady, supersonic, three-dimensional, frozen-inhomogeneous or ideal gas flows; however, the program was written in such a way that the extension to more complicated thermochemical models is relatively easily accomplished. A discussion of the problems encountered in writing the program and some typical results for the flow fields around conical and elliptical afterbodies are given.

A description of the machine program and comments concerning the functions of each subroutine are presented in Appendix A. Flow diagrams, output description and input and operating procedures are also included.

Section 2

DERIVATIONS OF GENERAL EQUATIONS

In this section the equations which are required to solve a general nonequilibrium, three-dimensional steady flow problem numerically, utilizing the method of characteristics, are derived. The general theory of the method of characteristics as it applies to a system of first order hyperbolic partial differential equations is first developed. Then, the general equations governing the flow of a reacting and relaxing gas are presented. Finally, the theory will be applied to the given governing equations in order to determine the characteristic conditions.

2.1 THE METHOD OF CHARACTERISTICS

The general theory for the method of characteristics is well developed and may be considered classical. Only that part of the theory which is necessary for the development of the proposed numerical scheme will be presented here. A more complete development may be found in many works on partial differential equations among which the more notable are Courant and Hilbert[36] and Hadamard[37].

A general system of n -first order, quasi-linear partial differential equations in n -dependent variables u_ν and m -independent variables x_i is considered. Using the index summation notation of cartesian tensors, these can be written in the form

$$a_{\mu\nu i} \frac{\partial u_\nu}{\partial x_i} = b_\mu \tag{1}$$

where the $a_{\mu\nu i}$ and b_μ are given functions of the u_ν and x_i . Quasi-linear systems of equations are those in which the highest order derivatives appear linearly. The restriction here to first order equations does not imply any loss of generality, because systems containing higher order derivatives can be reduced to a system of first order equations by defining new dependent variables.

The method of characteristics in two-independent variables which transforms the governing partial differential equations into ordinary differential equations along certain characteristic directions is a very special case of the general theory, and hence it cannot be easily generalized. However, the basic property that characteristic curves in two-variables are curves along which the derivatives are continuous but across which derivative discontinuities may occur can also be taken as the definition of a characteristic in m -independent variables. This property is used in two-dimensions in order to identify the characteristics as paths of waves in the physical model. In general then, a characteristic in m -independent variables can be defined as a subspace of $m-1$ -dimensions along which the derivatives of the dependent variables are continuously differentiable but across which discontinuities in the variables are allowed to occur. A characteristic surface (or hypersurface in more than three-independent variables) is thus once again associated with the surface generated by a wave front.

If the dependent variables u_ν are given as smooth functions on a characteristic surface, it then follows that the "interior" derivatives, that is, derivatives tangential to a characteristic are continuous and only the "exterior" derivatives (derivatives normal to the surface) may be discontinuous.

We consider a characteristic surface whose equation is

$$\tilde{f}(x_i) = 0 \tag{2}$$

and apply a transformation to the m -independent characteristic variables x_i' by taking

$$x_1' = \tilde{f}(x_i) \; ; \quad x_2' = \tilde{g}(x_i) \; , \quad x_3' = \tilde{h}(x_i) \; ; \quad x_4' = \tilde{k}(x_i) \tag{3}$$

Here, $m = 4$ and the functions $\tilde{g}$, $\tilde{h}$ and $\tilde{k}$ are arbitrary provided they form an independent set with $\tilde{f}$. At the characteristic $x_1' = 0$ the derivatives $\frac{\partial}{\partial x_2'}$, $\frac{\partial}{\partial x_3'}$ and $\frac{\partial}{\partial x_4'}$ are "interior" to the characteristic and are therefore continuous (since u_ν are smooth). Thus, only the derivatives $\frac{\partial u_\nu}{\partial x_1'}$ may be discontinuous.

Now, the derivative of any dependent variable u_ν with respect to one of the original independent variables is related to its derivatives with respect to the new variables (x_i') by the equations,

$$\frac{\partial u_\nu}{\partial x_i} = \frac{\partial x_j'}{\partial x_i} \frac{\partial u_\nu}{\partial x_j'} \tag{4}$$

The jump in $\frac{\partial u_\nu}{\partial x_i}$ at $x_1' = 0$ is therefore

$$\left[\frac{\partial u_\nu}{\partial x_i}\right] = \left[\frac{\partial u_\nu}{\partial x_1'}\right]\frac{\partial x_1'}{\partial x_i} = \left[\frac{\partial u_\nu}{\partial x_1'}\right]\frac{\partial \tilde{f}}{\partial x_i} \tag{5}$$

since all other derivatives are continuous as mentioned above. If (5) is substituted into the set of equations (1) a set of linear homogeneous equations for $\left[\frac{\partial u_\nu}{\partial x_1'}\right]$ with coefficients depending upon the $\frac{\partial x_1'}{\partial x_i}$ and the dependent variables u_ν is obtained as

$$\left[\frac{\partial u_\nu}{\partial x_1'}\right] a'_{\mu\nu 1} = 0 \tag{6}$$

where the $a'_{\mu\nu i}$ are the transformed coefficients of $\frac{\partial u_\nu}{\partial x_i}$ which are known functions of the u_ν and x_i'. Not all of the jumps $\left[\frac{\partial u_\nu}{\partial x_i}\right]$ can be zero if a genuine discontinuity is possible. Hence, the determinant of the coefficient array $a'_{\mu\nu i}$ must vanish, giving a relationship between the coefficients which must be satisfied if $\tilde{f}(x_i) = 0$ is to be a characteristic. Thus, the characteristic surfaces (or hypersurfaces) are defined by

$$\det\left\{a'_{\mu\nu 1}\right\} = \det\left\{a_{\mu\nu i}\left(\frac{\partial x_1'}{\partial x_i}\right)\right\} = 0 \tag{7}$$

Equation (7) is generally termed the characteristic equation for the characteristic $x_1' =$ constant.

The vanishing of the coefficient determinant (7) can be seen to have another implication. Writing the original equations (1) in terms of the new independent variables (x_1', x_2', x_3', x_4', $\cdots$) a set of equations is obtained which will have the form

$$A_{\mu\nu}(\tilde{f})\frac{\partial u_\nu}{\partial x_1'} = -A_{\mu\nu}(\tilde{g})\frac{\partial u_\nu}{\partial x_2'} - A_{\mu\nu}(\tilde{h})\frac{\partial u_\nu}{\partial x_3'} + \cdots + B_\mu . \tag{8}$$

Where, the functional form

$$A_{\mu\nu}(\phi) = a_{\mu\nu i}\frac{\partial \phi}{\partial x_i} \tag{9}$$

is the matrix of the system of transformed equations corresponding to (1) when $\tilde{f}$ is replaced by ϕ, and the vector B_μ contains all of the terms in the original equations which do not contain derivatives.

Now, if $\tilde{f}(x_i)$ is to be a characteristic manifold then by (7) the determinant of $A_{\mu\nu}(\tilde{f})$ must vanish. Thus the rows of $A_{\mu\nu}(\tilde{f})$ must be linearly dependent and there exists a vector h_μ such that

$$h_\mu A_{\mu\nu}(\tilde{f}) = 0 \quad \text{on} \quad \tilde{f}(x_i) = 0 \tag{10}$$

If Equations (8) are multiplied by this vector a linear combination of the equations is obtained which does not involve derivatives with respect tc x_1' as

$$h_\mu A_{\mu\nu}(\tilde{g})\frac{\partial u_\nu}{\partial x_2'} + h_\mu A_{\mu\nu}(\tilde{h})\frac{\partial u_\nu}{\partial x_3'} + \cdots = h_\mu B_\mu \tag{11}$$

Equation (11) is called the characteristic or compatibility condition and does not involve derivatives with respect to the coordinate x_1', i.e., in the direction normal to a characteristic surface. There may be more than one such equation associated with a particular characteristic surface, but there must always be at least one. Reference 38 contains a discussion of the possible number of compatibility equations associated with a given characteristic manifold. The compatibility equations are fundamental to the method of characteristics and will be derived below for the cases of three-dimensional steady and unsteady fluid flow.

2.2 THE EQUATIONS OF MOTION

The equations of conservation of species, mass, momentum and energy along with an equation of state govern a three-dimensional nonsteady, nonequilibrium gas flow. Neglecting all transport effects (viscosity, heat conduction, etc.) these equations in non-dimensional form become:

Continuity Equation

$$\frac{\partial \rho}{\partial t} + u\frac{\partial \rho}{\partial x} + v\frac{\partial \rho}{\partial y} + w\frac{\partial \rho}{\partial z} + \rho\left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z}\right) = 0 \tag{12}$$

x - Momentum

$$\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y} + w\frac{\partial u}{\partial z} + \frac{1}{\rho}\frac{\partial p}{\partial x} = 0 \tag{13}$$

y - Momentum

$$\frac{\partial v}{\partial t} + u\frac{\partial v}{\partial x} + v\frac{\partial v}{\partial y} + w\frac{\partial v}{\partial z} + \frac{1}{\rho}\frac{\partial p}{\partial y} = 0 \tag{14}$$

z - Momentum

$$\frac{\partial w}{\partial t} + u\frac{\partial w}{\partial x} + v\frac{\partial w}{\partial y} + w\frac{\partial w}{\partial z} + \frac{1}{\rho}\frac{\partial p}{\partial z} = 0 \tag{15}$$

Energy Equation

$$\frac{\partial h}{\partial t} + u\frac{\partial h}{\partial x} + v\frac{\partial h}{\partial y} + w\frac{\partial h}{\partial z} - \frac{1}{\rho}\left(\frac{\partial p}{\partial t} + u\frac{\partial p}{\partial x} + v\frac{\partial p}{\partial y} + w\frac{\partial p}{\partial z}\right) = 0 \tag{16}$$

Conservation of i th Species

$$\frac{\partial \gamma_i}{\partial t} + u\frac{\partial \gamma_i}{\partial x} + v\frac{\partial \gamma_i}{\partial y} + w\frac{\partial \gamma_i}{\partial z} = \frac{Q_i}{\rho} \qquad i = E+1, \cdots, NS \tag{17}$$

Conservation of i th Element

$$\sum_{i=1}^{NS} \alpha_{ik}\left(\frac{\partial \gamma_i}{\partial t} + u\frac{\partial \gamma_i}{\partial x} + v\frac{\partial \gamma_i}{\partial y} + w\frac{\partial \gamma_i}{\partial z}\right) = 0 \qquad k = 1, 2, \cdots, E \tag{18}$$

Vibrational Energy Equation

$$\frac{\partial \epsilon_j}{\partial t} + u\frac{\partial \epsilon_j}{\partial x} + v\frac{\partial \epsilon_j}{\partial y} + w\frac{\partial \epsilon_j}{\partial z} = R_{v_j} \qquad j = 1, NV \tag{19}$$

Equations of State

$$h = \sum_{i=1}^{NS} h_i \gamma_i \tag{20}$$

and

$$\Lambda p = \rho T \sum_{i=1}^{NS} \gamma_i \tag{21}$$

Equations (12)-(21) represent $NS+NV+7$ equations in $NS+NV+7$ unknowns with $NS+NV-E$ functions (Q_i , R_{v_j}) to be specified and thus form a complete set. Most of the thermodynamic and chemical variables have been non-dimensionalized by using their free-stream counterparts. The exceptions to this are the pressure which is normalized with respect to $\rho'_\infty U'^2_\infty$ and the energies for which $R'T'_\infty$ is used. Distances are normalized with respect to a characteristic length L' (which in the case of a blunted body is usually the nose radius of curvature), and time with respect to $\frac{L'}{U'_\infty}$. In the Equation of State (21), Λ is the normalizing term

$$\Lambda = \frac{(u'_\infty)^2 MW'_\infty}{R'T'_\infty} \tag{22}$$

The gas in the shock layer has been assumed to consist of NS monatomic and diatomic species M_j connected by the R reactions

$$\sum_{j=1}^{NS} \nu_{ij} M_j \underset{k'_{bi}}{\overset{k'_{fi}}{\rightleftharpoons}} \sum_{j=1}^{NS} \nu_{ij}^{*} M_j \qquad i = 1, 2, \cdots, NR \qquad (23)$$

where ν_{ij} and ν_{ij}^{*} represent the stoichiometric coefficients. Concentrations are expressed in mole fractions referred to the free stream molecular weight or

$$\gamma_i = \frac{\gamma_i'}{\sum_{i=1}^{NS} \gamma_{i\infty}'} = \gamma_i' \, MW_\infty' \qquad (24)$$

Equations (17) and (18) govern the rate of change of the concentration of the i th species. Here, α_{ik} is the number of atoms of the k th element per molecule of the i th species, E represents the number of elements and NS represents the total number of species. Equation (21) governs the rate of vibrational energy relaxation where ϵ_j is the vibrational energy for each of the NV species which are not in vibrational equilibrium.

The source terms Q_i and R_{V_j} are complicated functions of the temperature, density and the rates of certain reactions involved in the species i and j. Since the exact forms of the source terms are not necessary for a development of the proposed numerical solution, a full discussion of them shall not be given here. References (39) and (33) describe the source terms which are currently being used for non-equilibrium flow field calculations.

The enthalpy of the j th species used in Equation (20) and the other species thermodynamic quantities are generally calculated from a single harmonic oscillator approximation to the diatomic molecules or from curve fits as functions of temperature. Here, the caloric equation of state of the i th species is given as

$$h_i = I_i + h_i^{\circ} + \epsilon_i \qquad (25)$$

where I_i includes the energy of rotation and translation, h_i° the heat of formation and ϵ_i the energy of vibration. Equations (25), (20) and (21) can be used to eliminate the derivatives of ρ in Equation (12) Differentiating Equation (20) gives

$$\frac{dh}{dt} = \sum_{i=1}^{NS} \gamma_i \frac{dh_i}{dt} + \sum_{i=1}^{NS} h_i \frac{d\gamma_i}{dt} \tag{26}$$

where $\frac{d}{dt} = \frac{\partial}{\partial t} + u \frac{\partial}{\partial x} + v \frac{\partial}{\partial y} + w \frac{\partial}{\partial z}$ is the "substantial" or Stokes derivative. Substituting for h_i from (25)

$$\frac{dh}{dt} = \sum_{i=1}^{NS} \gamma_i \left(\frac{\partial I_i}{\partial T}\right)\frac{dT}{dt} + \sum_{i=1}^{NS-NV} \gamma_i \left(\frac{\partial \epsilon_i}{\partial T}\right)\frac{dT}{dt} + \sum_{i=1}^{NS} h_i \frac{d\gamma_i}{dt} + \sum_{j=1}^{NV} \gamma_j \frac{d\epsilon_j}{dt} \tag{27}$$

Then by defining

$$C_p = \sum_{i=1}^{NS} \gamma_i \frac{\partial I_i}{\partial T} + \sum_{i=1}^{NS-NV} \gamma_i \frac{\partial \epsilon_i}{\partial T} \tag{28}$$

as the specific heat at constant pressure, Equation (27) becomes

$$\frac{dh}{dt} = C_p \frac{dT}{dt} + \sum_{i=1}^{NS} h_i \frac{d\gamma_i}{dt} + \sum_{j=1}^{NV} \gamma_j \frac{d\epsilon_j}{dt} \tag{29}$$

The equation of state (21) can be used to eliminate $\frac{dT}{dt}$ from Equation (29) Differentiating (21) and combining the result with (29) gives

$$\frac{1}{\rho}\frac{C_p MW - 1}{C_p \Re T}\frac{dp}{dt} - \frac{1}{\rho}\frac{d\rho}{dt} - MW \sum_{i=1}^{NS} \frac{d\gamma_i}{dt} = -\left(\frac{1}{C_p T}\right)\left[\sum_{i=1}^{NS} \frac{Q_i h_i}{\rho} + \sum_{i=1}^{NS} \gamma_i \Re' v_i\right] \tag{30}$$

where $\Re \equiv \frac{1}{\Lambda}$ and $MW = \frac{1}{\sum_{i=1}^{NS} \gamma_i}$. (31)

Eliminating the $\frac{1}{\rho}\frac{d\rho}{dt}$ term from (12) and (30) gives finally

$$\bar{\nabla}\cdot\bar{q} + \frac{1}{\rho a^2}\frac{d\rho}{dt} = -q\bar{f} \tag{32}$$

where

$$\bar{f} = \frac{1}{qC_pT}\left(\sum_{j=1}^{NV}\gamma_j R v_j + \sum_{i=1}^{NS}\frac{Q_i h_i}{\rho}\right) - MW\sum_{i=1}^{NS}\frac{Q_i}{\rho q} \tag{33}$$

contains all the nonequilibrium rate terms from Equations (17) and (19) and

$$a^2 = \frac{C_p \mathcal{R} T}{C_p MW - 1} \tag{34}$$

is the frozen speed of sound. Chu[25] has shown that the frozen sound speed a, defined by (34), is the wave velocity of a flow which is not in equilibrium and thus, can be associated with the rate of propagation of characteristic surfaces. This result will be derived in the next section. Equations (13), (19) and (33) become the basic equations of change to which the theory of characteristics developed in the last section will be applied.

2.3 THE CHARACTERISTIC RELATIONS

Here, the theory presented in Paragraph 2.2 is applied to the flow equations given in Paragraph 2.1. The governing equations for steady flow are obtained from Equations (12) through (21) by setting the terms involving $\frac{\partial}{\partial t}$ identically equal to zero.

Applying Equation (5) to the governing flow equations presented in Paragraph 2.2 leads to the following homogeneous equations for the jump conditions at a characteristic surface

$$x_1' = 0 \tag{35}$$

$$\frac{1}{\rho a^2}\left(u\tilde{f}_x + v\tilde{f}_y + w\tilde{f}_z\right)\left[\frac{\partial p}{\partial x_1'}\right] + \tilde{f}_x\left[\frac{\partial u}{\partial x_1'}\right] + \tilde{f}_y\left[\frac{\partial v}{\partial x_1'}\right] + \tilde{f}_z\left[\frac{\partial w}{\partial x_1'}\right] = 0 \tag{36}$$

$$\frac{1}{\rho}\tilde{f}_x\left[\frac{\partial p}{\partial x_1'}\right] + \left(u\tilde{f}_x + v\tilde{f}_y + w\tilde{f}_z\right)\left[\frac{\partial u}{\partial x_1'}\right] = 0 \tag{37}$$

$$\frac{1}{\rho}\tilde{f}_y\left[\frac{\partial p}{\partial x_1'}\right] + \left(u\tilde{f}_x + v\tilde{f}_y + w\tilde{f}_z\right)\left[\frac{\partial v}{\partial x_1'}\right] = 0 \tag{38}$$

$$\frac{1}{\rho}\tilde{f}_z\left[\frac{\partial p}{\partial x_1'}\right] + \left(u\tilde{f}_x + v\tilde{f}_y + w\tilde{f}_z\right)\left[\frac{\partial w}{\partial x_1'}\right] = 0 \tag{39}$$

$$\left(u\tilde{f}_x + v\tilde{f}_y + w\tilde{f}_z\right)\left[\frac{\partial \gamma_i}{\partial x_1'}\right] = 0 \qquad i = E+1, E+2, \cdots, NS \tag{40}$$

$$\left(u\tilde{f}_x + v\tilde{f}_y + w\tilde{f}_z\right)\left[\frac{\partial \epsilon_j}{\partial x_1'}\right] = 0 \qquad j = 1, 2, \cdots, NV \tag{41}$$

The vanishing of the determinant of the coefficients (Equation 7) implies that for $x_1' = 0$ to be a characteristic surface the condition

$$\left(u\tilde{f}_x + v\tilde{f}_y + w\tilde{f}_z\right)^{NS-E+NV+2}\left\{\left(u\tilde{f}_x + v\tilde{f}_y + w\tilde{f}_z\right)^2 - a^2\left(\tilde{f}_x^2 + \tilde{f}_y^2 + \tilde{f}_z^2\right)\right\} = 0 \tag{42}$$

must be satisfied. It can be seen from Equation (42) that there are two sets of characteristic surfaces corresponding to each of the two distinct factors.

Each factor in (42) corresponds to conditions for the directions of the normals ($\tilde{f}_x$, $\tilde{f}_y$, $\tilde{f}_z$) to the respective characteristic surfaces. The first factor is the dot product of the velocity vector

$$\bar{q} = u\bar{i} + v\bar{j} + w\bar{k} \tag{43}$$

and the vector normal to the characteristic surfaces $\tilde{f}(x, y, z) = 0$

$$\nabla\tilde{f} = \frac{\partial\tilde{f}}{\partial x}\bar{i} + \frac{\partial\tilde{f}}{\partial y}\bar{j} + \frac{\partial\tilde{f}}{\partial z}\bar{k} \tag{44}$$

Hence, the vanishing of this product implies that the surfaces composed of streamlines are $NS - E + NV + 2$ - fold characteristics corresponding to the power of the term. This fact will prove to be very useful in carrying out a numerical integration scheme for the full nonequilibrium equations.

The second factor of (42) implies that the normal vector $(\tilde{f}_x, \tilde{f}_y, \tilde{f}_z)$ to the surface

$$\tilde{f}(x, y, z) = \text{constant} \tag{45}$$

lies on a cone of the second degree given by

$$a_{ik}\, x_i' x_k' = (ux_1' + vx_2' + wx_3')^2 - a^2(x_1'^2 + x_2'^2 + x_3'^2) = 0 \tag{46}$$

Equation (46) can be shown to be real and thus the governing equations are hyperbolic whenever $q^2 = (u^2 + v^2 + w^2) > a^2$ i.e. when the flow is supersonic.

Any displacement (dx, dy, dz) along the characteristic surface (45) through a point P can be given in the limit by

$$\tilde{f}_x\, dx + \tilde{f}_y\, dy + \tilde{f}_z\, dz = 0 \tag{47}$$

and it follows that if ($\tilde{f}_x$, $\tilde{f}_y$, $\tilde{f}_z$) lies on the normal cone (46) through P, then the envelope of all such displacements through P is itself a

quadratic cone. This envelope, which is called the "characteristic conoid" at P is given by (See Courant and Hilbert P. 563)[36] inverting the quadratic form (46) as

$$a_{ij}^{-1} dx_i dx_j = (v^2+w^2-a^2)dx^2 + (u^2+w^2-a^2)dy^2 + (u^2+v^2-a^2)dz^2 \tag{48}$$

$$- 2uv\,dx\,dy - 2uw\,dx\,dz - 2vw\,dy\,dz = 0$$

where a_{ij}^{-1} is the inverse of a_{ij} in (46). Equation (48) can also be shown to be the equation of the well known Mach conoid which usually appears in supersonic flow. In general, it represents a curvilinear cone, whose coefficients are functions of the coordinates (x, y, z) but whose generators are tangent to the generators of the local Mach cone at P. In any numerical procedure, the characteristic conoid is replaced by an average approximation to the local Mach cone.

The generators dx_i (called the bicharacteristics) of the local Mach cone (48) can be expressed parametrically by[21]

$$dx_i = (u_i + C\alpha_i \cos\delta + C\beta_i \sin\delta)\,d\tau \tag{49}$$

where δ is a parametric angle, α_i, β_i are an orthogonal set of vectors, C is defined by

$$C^2 = \frac{q^2a^2}{q^2-a^2} = \frac{q^2}{M^2-1} \tag{50}$$

and τ is the time taken for propagation of a wave along a ray corresponding to a bicharacteristic.

To determine the α_i and β_i, we now consider a special intrinsic coordinate system of the type originally proposed by Thornhill (Figure 1). Here, σ is the angle made by the plane containing the lines $\bar{q}$ and $\bar{l}$ with the plane containing the lines $\bar{q}$ and y. $\bar{l}$ is a ray of the characteristic conic (Mach surface) making an angle β with the velocity vector $\bar{q}$ and

θ, ψ are the polar angles of the velocity vector. The coordinate system is chosen aligned as shown with $\bar{\xi}$ along $\bar{q}$, $\bar{\eta}$ in the plane of ψ and $\bar{q}$ perpendicular to $\bar{q}$ and $\bar{\zeta}$ perpendicular to the plane of ψ and $\bar{q}$. $\bar{\xi}, \bar{\eta}$ and $\bar{\zeta}$ are chosen to be unit vectors and are given by

$$\bar{\xi} = \cos\theta\cos\psi\bar{i} + \sin\theta\bar{j} + \cos\theta\sin\psi\bar{k} = \frac{1}{q}(u\bar{i}+v\bar{j}+w\bar{k}) \tag{51}$$

$$\bar{\eta} = -\sin\theta\cos\psi\bar{i} + \cos\theta\bar{j} - \sin\theta\sin\psi\bar{k} = \alpha_1\bar{i} + \alpha_2\bar{j} + \alpha_3\bar{k} \tag{52}$$

$$\bar{\zeta} = -\sin\psi\bar{i} + \cos\psi\bar{k} = \beta_1\bar{i} + \beta_2\bar{j} + \beta_3\bar{k} \tag{53}$$

Therefore, in the form of (49) the bicharacteristic direction $\bar{L}$ is given by

$$\begin{aligned} dx &= (u - C\sin\theta\cos\psi\cos\delta - C\sin\psi\sin\delta)d\tau \\ dy &= (v + C\cos\theta\cos\delta)d\tau \\ dz &= (w - C\sin\psi\sin\theta\cos\delta + C\cos\psi\sin\delta)d\tau \end{aligned} \tag{54}$$

2.4 THE COMPATIBILITY EQUATIONS

As shown in Paragraph 2.1, the compatibility equations are determined by forming a linear combination of the flow equations and choosing the coefficients such that the derivatives in the direction normal of the characteristic surface disappear. There are two sets of compatibility equations corresponding to the two sets of characteristic surfaces which were described in the last section. Actually since the streamlines are $NS-E+NV+2$-fold characteristics, there are $NS-E+NV+2$ compatibility equations which can be found to apply along them.

The equations which apply along the streamline surfaces could be found by applying the method developed in 2.1 to the full equations of motion. However, in this case, it is easier to recall that the equations sought do not contain derivatives in the direction normal to the streamline. This property can be used to choose the required relations by inspection

from the original partial differential equations. The streamline directions are given by

$$d\bar{\xi} = \bar{q}\, dt \tag{55}$$

where $\bar{\xi}$ is the coordinate along the streamline. Thus, $NV-E+NS$ compatibility equations can be found by transforming Equations (17) and (19) to directional derivatives along the streamline. The equations become

$$\frac{d\gamma_i}{d\xi} = \frac{Q_i}{\rho q} \qquad i = E+1, \cdots, NS \tag{56}$$

and

$$\frac{de_j}{d\xi} = \frac{R v_j}{q} \qquad j = 1, 2, \cdots, NV \tag{57}$$

By introducing Equation (20) for the enthalpy h into the energy equation (16) and differentiating in the manner used to obtain Equation (29), the energy quotation along a streamline can be written as

$$C_p \frac{dT}{d\xi} + \frac{1}{q}\left(\sum_{i=1}^{NS} \frac{h_i Q_i}{\rho} + \sum_{j=1}^{NV} \gamma_j R v_j\right) - \frac{\Lambda}{\rho}\frac{d\rho}{d\xi} = 0 \tag{58}$$

One further relation is required. This is found from scalar multiplication of the momentum equations and the velocity vector $\bar{q}$ as

$$\frac{d}{d\xi}\left(\frac{1}{2} q^2\right) = -\frac{1}{\rho}\left(\frac{d\rho}{d\xi}\right) = -\frac{1}{\Lambda}\frac{dh}{d\xi} \tag{59}$$

where the last term on the right hand side of (59) is obtained from the energy equation (16). Since Equations (56) - (59) contain no derivatives in the direction normal to the streamlines, they are the required $NS-E+NV+2$ compatibility equations.

The compatibility equation which applies along the characteristic conoid (48) can be found by applying the method of Paragraph 2.1 to Equations (13) and (32). Equations (35) - (39) are representative of the equations of motion transformed to characteristic coordinates (x_1', x_2', x_3'). Multiplying these equations by h_1, h_2, h_3 and h_4 respectively leads to the following equation for the term $h_\mu A_{\mu\nu}(f) \frac{\partial u_\nu}{\partial x_1'}$ presented in Paragraph 2.1.

$$\frac{1}{\rho}\left(\frac{1}{a^2} h_1 (\bar{q}\cdot\nabla\tilde{f}) + h_2\tilde{f}_x + h_3\tilde{f}_y + h_4\tilde{f}_z\right)\left[\frac{\partial p}{\partial x_1'}\right] + \left(h_2(\bar{q}\cdot\nabla\tilde{f}) + h_1\tilde{f}_x\right)\left[\frac{\partial u}{\partial x_1'}\right]$$
$$+ \left(h_3(\bar{q}\cdot\nabla\tilde{f}) + h_1\tilde{f}_y\right)\left[\frac{\partial v}{\partial x_1'}\right] + \left(h_4(\bar{q}\cdot\nabla\tilde{f}) + h_1\tilde{f}_z\right)\left[\frac{\partial w}{\partial x_1'}\right] = h_\mu A_{\mu\nu}(\tilde{f})\frac{\partial u_\nu}{\partial x_1'} \tag{60}$$

The condition for which the surface $\tilde{f}(x,y,z) = 0$ is to be a characteristic surface is that

$$h_\mu A_{\mu\nu}(\tilde{f}) = 0 \tag{61}$$

Hence, the coefficients of $\frac{\partial u_\nu}{\partial x_i}$ must all vanish identically. Equation (60) thus leads to the following expressions for the linear factors h_μ .

$$\frac{h_2}{h_1} = -\left(\frac{\partial\tilde{f}}{\partial x}\right)\Big/\bar{q}\cdot\nabla\tilde{f} \tag{62}$$

$$\frac{h_3}{h_1} = -\left(\frac{\partial\tilde{f}}{\partial y}\right)\Big/\bar{q}\cdot\nabla\tilde{f} \tag{63}$$

$$\frac{h_4}{h_1} = -\left(\frac{\partial\tilde{f}}{\partial z}\right)\Big/\bar{q}\cdot\nabla\tilde{f} \tag{64}$$

Using these values for h_μ , the compatibility equation corresponding to Equation (11) becomes

$$\sum_{i=2}^{3}\left\{\frac{1}{\rho}\left[\frac{1}{a^2}\bar{q}\cdot\nabla\phi - \left(\frac{\partial\tilde{f}}{\partial x}\Big/\bar{q}\cdot\nabla\tilde{f}\right)\phi_x - \left(\frac{\partial\tilde{f}}{\partial y}\Big/\bar{q}\cdot\nabla\tilde{f}\right)\phi_y - \left(\frac{\partial\tilde{f}}{\partial z}\Big/\bar{q}\cdot\nabla\tilde{f}\right)\phi_z\right]\frac{\partial p}{\partial x_i'}\right.$$
$$\left. + \left(\phi_x - \tilde{f}_x\frac{\bar{q}\cdot\nabla\phi}{\bar{q}\cdot\nabla\tilde{f}}\right)\frac{\partial u}{\partial x_i'} + \left(\phi_y - \tilde{f}_y\frac{\bar{q}\cdot\nabla\phi}{\bar{q}\cdot\nabla\tilde{f}}\right)\frac{\partial v}{\partial x_i'} + \left(\phi_z - \tilde{f}_z\frac{\bar{q}\cdot\nabla\phi}{\bar{q}\cdot\nabla\tilde{f}}\right)\frac{\partial w}{\partial x_i'}\right\} = -q\tilde{f} \tag{65}$$

where $\phi = \tilde{q}, \tilde{h}$ for $i = 2, 3$ respectively.

Equation (65) can be simplified by choosing as the coordinates (x_1', x_2', x_3') the basic characteristic coordinate system. Referring again to Figure 1, take as the coordinates $\bar{L}$, $\bar{N}$ and $\bar{M}$ where $\bar{L}$ is along the bicharacteristic given by δ, $\bar{N}$ is orthogonal to $\bar{L}$ in the plane of $\bar{\eta}$ and $\bar{\zeta}$ and $\bar{M} = \bar{L} \times \bar{N}$ is orthogonal to the characteristic surface and positive outward. Here, $(x_1', x_2', x_3') = (M, L, N)$ and

$$\tilde{f} = \bar{M} = \cos\beta\cos\delta\bar{\eta} + \cos\beta\sin\delta\bar{\zeta} - \sin\beta\bar{\xi} \tag{66}$$

$$\tilde{g} = \bar{L} = \sin\beta\cos\delta\bar{\eta} + \sin\beta\sin\delta\bar{\zeta} + \cos\beta\bar{\xi} \tag{67}$$

$$\tilde{h} = \bar{N} = \sin\delta\bar{\eta} - \cos\delta\bar{\zeta} \tag{68}$$

$$\text{where } \beta = \sin^{-1}\frac{1}{M} \tag{69}$$

With respect to the characteristic coordinate system (46) - (48) Equation (65) becomes

$$\begin{aligned}
&\frac{1}{\rho a}\frac{\partial p}{\partial L} + (-\sin\beta\cos\delta\cos\theta\cos\psi - \sin\beta\sin\delta\sin\psi - \cos^2\beta\cos\delta\sin\theta\cos\psi/\sin\beta \\
&\quad - \cos^2\beta\sin\delta\sin\psi/\sin\beta)\frac{\partial u}{\partial L} + (\cos\delta\sin\psi - \sin\delta\sin\theta\cos\psi)\frac{\partial u}{\partial M} \\
&\quad + (\sin\beta\cos\delta\cos\theta + \cos^2\beta\cos\delta\cos\theta/\sin\beta)\frac{\partial v}{\partial L} + \sin\delta\cos\theta\frac{\partial v}{\partial M} \\
&\quad + (\sin\beta\sin\delta\cos\psi - \cos\delta\sin\theta\sin\psi\sin\beta \\
&\quad\quad - \cos^2\beta\cos\delta\sin\theta\sin\psi/\sin\beta + \cos^2\beta\sin\delta\cos\psi/\sin\beta)\frac{\partial w}{\partial L} \\
&\quad + (-\sin\delta\sin\theta\sin\psi - \cos\delta\cos\psi)\frac{\partial w}{\partial M} = -q\tilde{x}
\end{aligned} \tag{70}$$

Changing from the variables (p, u, v, w) to the variables (p, q, θ, ψ) and simplifying, the final form of the compatibility equation becomes

$$\begin{aligned}
&\frac{\cot\beta}{\rho q^2}\frac{\partial p}{\partial L} + \cos\delta\frac{\partial\theta}{\partial L} + \cos\theta\sin\delta\frac{\partial\psi}{\partial L} \\
&\quad = -\sin\beta\left(\cos\delta\cos\theta\frac{\partial\psi}{\partial N} - \sin\delta\frac{\partial\theta}{\partial N} + \tilde{x}\right)
\end{aligned} \tag{71}$$

where $\frac{\partial}{\partial L}$ and $\frac{\partial}{\partial N}$ are the derivatives along and normal to the

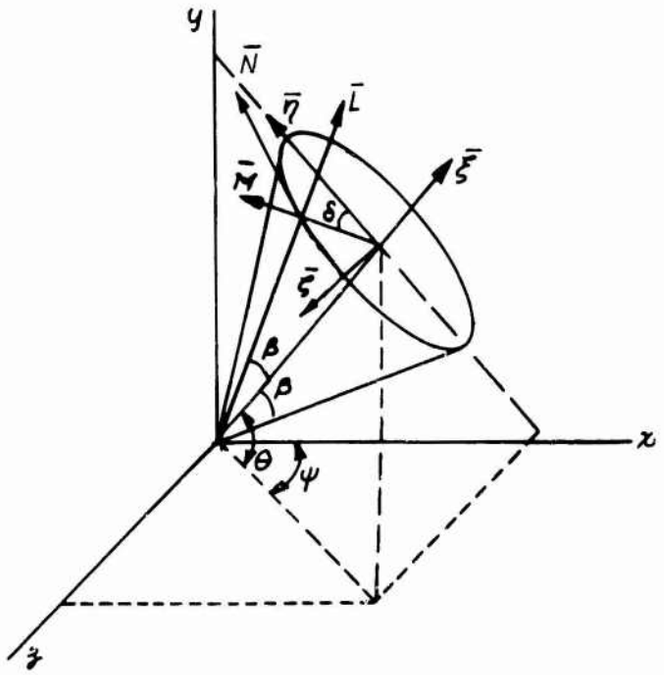


Figure 1 BASIC COORDINATE SYSTEM
3-D STEADY FLOW

bicharacteristic δ . Note that Equation (71) contains derivatives of ρ, θ and ψ only. Equations (56), (57), (58) and (71) are the basic set of equations which will be used to develop a numerical integration scheme for three-dimensional steady flow in Section 3.

Section 3
THE NUMERICAL SOLUTION

In this section, the details of a multivariable numerical integration scheme, which is based upon the method of characteristics applied to the equations developed in Section 2, are presented. First, the general requirements for the stability and convergence of such a scheme are considered, and then several possible difference networks are discussed and compared. Finally, the details of the proposed techniques are given for both steady and nonsteady nonequilibrium flows.

3.1 STABILITY AND COVERGENCE

In all numerical approaches, the solutions to a system of partial differential equations are represented by values of the dependent variables for certain discrete values of the independent variables. In general, using this procedure, there are three types of errors which would cause the numerical values to be different from those which would be obtained from the exact solution of the partial differential equations themselves. These can be classified as truncation errors, round off errors, and errors which appear in the initial values. The first is the error which results from using a difference formula which is an approximation to the true equation and may be considered as the error incurred from representing an infinite series by a finite number of terms. The second arises from the need to use finite decimal numbers in the computation, while the last may occur because of inaccurate description of the initial and boundary conditions. In principle, the cumulative effects of these errors upon the solution can be kept under control by reducing the mesh spacing between grid points and carrying out the calculations with sufficiently high precision.

In practice, however, the cumulative departure from the exact solution for a fixed range of the independent variable will usually grow unboundedly as the mesh size approaches zero. This is because the precision required to sufficiently control the growth would far exceed the capacity of existing computers. If this growth occurs exponentially as the mesh size decreases, it is generally considered unmanageable for computational purposes, and the procedure is termed unstable.[7] The establishment of conditions necessary to assure stable solutions thus assumes a major role

in any numerical scheme. For a set of first-order linear partial differential equations in n-independent variables, Courant, Friedrichs, and Lewy,[40] in their classical paper, have shown that a necessary conditions for stability (i.e., the C-F-L stability condition) is that the domain of dependence of the partial differential equations (e.g., the circle $\xi^2 + \eta^2 = c(\xi)$ for the Mach cone in steady flow shown in Figure 1) be contained within the domain of dependence of the difference equations. This latter domain is usually called the "convex hull" of the difference scheme and is the region obtained by connecting with straight-line segments the outermost points used in solving the difference equations. Forsythe[7] indicates that, for a central differencing scheme using four initial points in three dimensions, or a comparable number in n-dimensions, the C-F-L conditions is both necessary and sufficient to guarantee stability, and thus also convergence of the difference scheme. Hohn[41] has also found this to be the case when simplical difference schemes (i.e., schemes where a minimum number of points are utilized in the initial plane) are used for solving linear equations.

While the equations of motion presented in Section 2.2 are not linear, it can be argued on physical grounds that the C-F-L stability criterion is also likely necessary for stability of a difference scheme based upon these equations. Consider, for instance, the simple case of supersonic flow parallel to the x-axis and assume that the four points $(0, +h_1, 0)$, $(0, -h_1, 0)$, $(0, 0, -h_2)$, and $(0, 0, h_2)$ comprise the points in the initial plane $x = 0$ to be used in a central differencing scheme for calculating the flow properties at a new point $(k, 0, 0)$, (See Figure 2). The domain of dependence of the difference equations is then the rhombus formed with the four points in the initial plane as vertices. If the C-F-L stability condition is not satisfied, the circle which is the intersection of the Mach forecone from $(k, 0, 0)$ with the initial plane (e.g., the domain of dependence of the differential equations) would not be contained within the rhombus. Hence, certain initial values (e.g., points within the circle but outside the rhombus) could be arbitrarily changed without affecting the solution of the difference equations. The departure of the difference solution from the solution of the differential equations could thus become very large and the scheme would be unstable.

Mathematically, Strang[63] has shown that the convergence of the nonlinear equations depends uniquely upon the stability of the linear equations for cases where the equations and their solution possess enough continuous derivatives. For many cases of fluid flow, where the gradients are not too high (e.g., excluding flow through shocks, etc.) these conditions would be expected to be met, and the C-F-L condition should be sufficient to insure stability using the simple differencing schemes discussed above.

Although the above discussion applies specifically to numerical solutions using a direct finite difference network, the physical arguments given in paragraph three can also be seen to hold when a method of characteristics approach is used. Now, the four vertex points of the rhombus (convex hull of difference equations) would represent base points of four bicharacteristics along which the equations are to be integrated. These points would now lie on the circle and the rhombus would thus be contained completely within the circle. Hence, when no other points are included, the scheme would necessarily be unstable because some of the initial data is neglected. In the technique described below, the over-all procedure utilizing a method-of-characteristics solution is stabilized by introducing initial data points outside the domain of dependence of the differential equations and obtaining the flow properties at the bicharacteristic base points by interpolation among these initial points along the lines employed by Sauerwein and Sussman,[41] Tsung,[22] and Butler and Talbot.[42] The extra points added now form the outer boundary of the convex hull of the characteristic difference network and the C-F-L stability condition can be modified to apply to this region.

3.2 CHOICE OF A PRATICAL INTEGRATION NETWORK

The method of characteristics has been applied extensively to solve two-dimensional flow-field problems. In this case, there are two characteristics which pass through any initial point. The compatibility equations are ordinary differential equations which can be written in finite difference form and used to solve for the flow variables at a new point. Two basic networks have been utilized for carrying out the numerical integration. In the first, as explained by Ferri[12] and shown in Figure 3 point P_3 is located as the intersection of two characteristics from initial

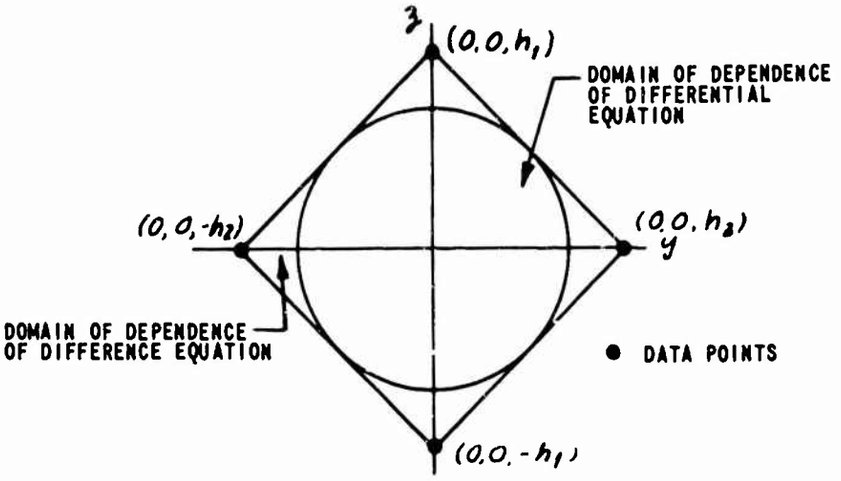


Figure 2 COURANT-FRIEDRICHS-LEWY CONDITION FOR FINITE DIFFERENCE SCHEME

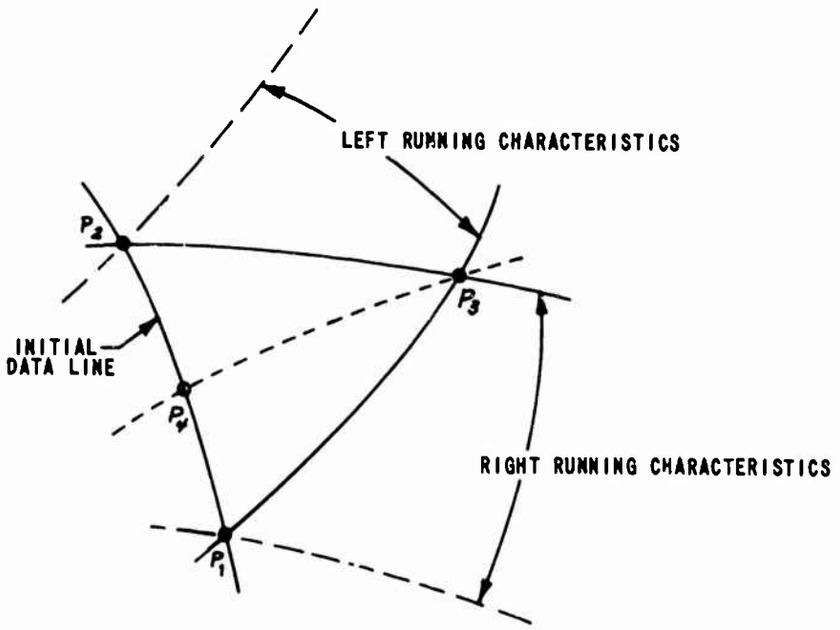


Figure 3 FORWARD CHARACTERISTIC NET FOR TWO INDEPENDENT VARIABLES

points P_1 and P_2, respectively. The flow variables at point P_3 are obtained by simultaneously solving the compatibility equations written along these characteristics. Here, compatibility equations along the streamline P_3 P_4 may be introduced as auxiliary conditions. The second method shown in Figure 4 was originally proposed by Hartree. This consists first of selecting the point P_3 off the initial line and intersecting the characteristics through P_3 with the initial line at points P_1 and P_2. The flow properties at points P_1 and P_2 are determined by interpolation and the finite difference forms of the bicharacteristic equation are used to solve for the flow variables at point P_3. Here again, the streamline through P_3 can be supplied when more than two dependent variables are involved.

Both methods described above for the two variable problem are essentially the same, because regardless of how point P_3 is located, the choice of characteristic directions through P_3 is not arbitrary. However, this is not the case for problems involving more than two independent variables. Indeed, as was shown in Equation (49), the generators of the characteristic cones in three-independent variables comprise a one-parameter family of curves through any point P. Thus, there is a certain degree of arbitrariness involved in choosing a basic finite difference network for use in a multivariable characteristic procedure.

Before discussing some of the networks which have thus far been proposed, it will first be useful to introduce the generalization of the two characteristic directions associated with any point P in two dimensions. For this purpose, consider an arbitrary curve that acts as a source of disturbance in three-space, (see Figure 5). A characteristic conoid is associated with each point on the curve in the manner indicated. The envelope of all these conoids, which also contains the original curve, forms two distinct surfaces (Σ_1 and Σ_2) which, since they are comprised of bicharacteristics, are also characteristic. Hence, through any curve in three space, there are two characteristic surfaces corresponding to the two characteristic lines through a point in two space. The geometry involved in the three-dimensional method of characteristics has thus become evident. Through each point P in space, the normal $\bar{N}$ to a characteristic surfact lies on the normal cone given by Equation (45) (see Figure 6).

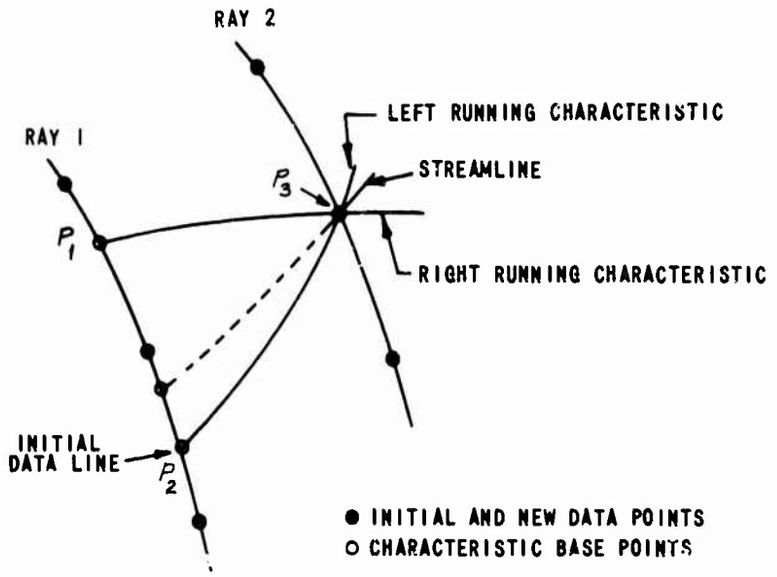


Figure 4 BACKWARD CHARACTERISTIC NET
TWO INDEPENDENT VARIABLES

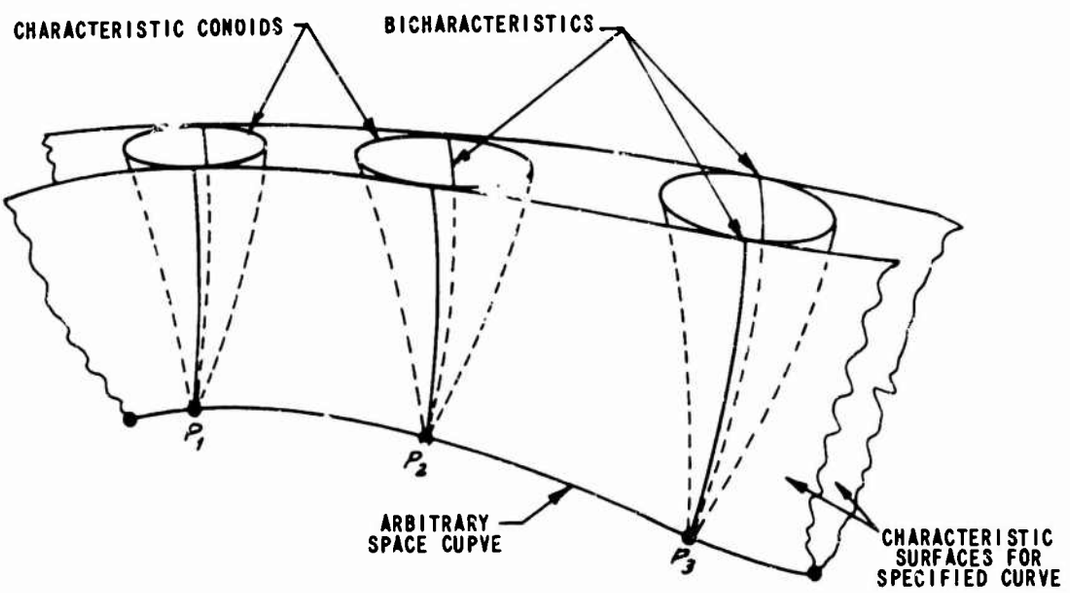


Figure 5 CHARACTERISTIC SURFACES THROUGH ARBITRARY CURVE

The envelope of all such surfaces through P is just the characteristic conoid at P and is itself a characteristic surface. The curves of contact between the characteristic conoid and the enveloping surfaces are the bicharacteristic curves. Thus, the characteristic conoid may be regarded as being generated by the bicharacteristic curves through P, its vertex. This characteristic conoid coincides with the Mach conoid at every point on its surface. As shown in Figure 7, the boundary of the Mach conoid is just the envelope of all the local infinitesimal Mach cones. The characteristic conoid is thus everywhere tangent to a local Mach cone.

The solution of flow field problems by the method of characteristics in more than two independent variables consists, therefore, of choosing from the infinity of directions available a particular set of bicharacteristics or characteristic surfaces, writing the equations developed in Section 2 along them in finite difference form and solving the resulting numerical equations simultaneously. Various finite difference networks have been proposed for accomplishing these integrations. The most important of these have been discussed and compared previously.[9,17] Each of the proposed schemes are summarized here for ease of reference and finally evaluated from the standpoint of their applicability to finite-rate chemically reacting flow field studies. In all of the procedures, the characteristic surfaces and conoids discussed above are replaced by suitable average planes and cones. The nomenclature used by Fowell and Sauerwein is repeated here.

Thornhill[14] originally proposed two integration schemes. In the first, termed the "tetrahedral bicharacteristic line network" (Figure 8), the new point P_4 is located as the common intersection point of Mach cones from

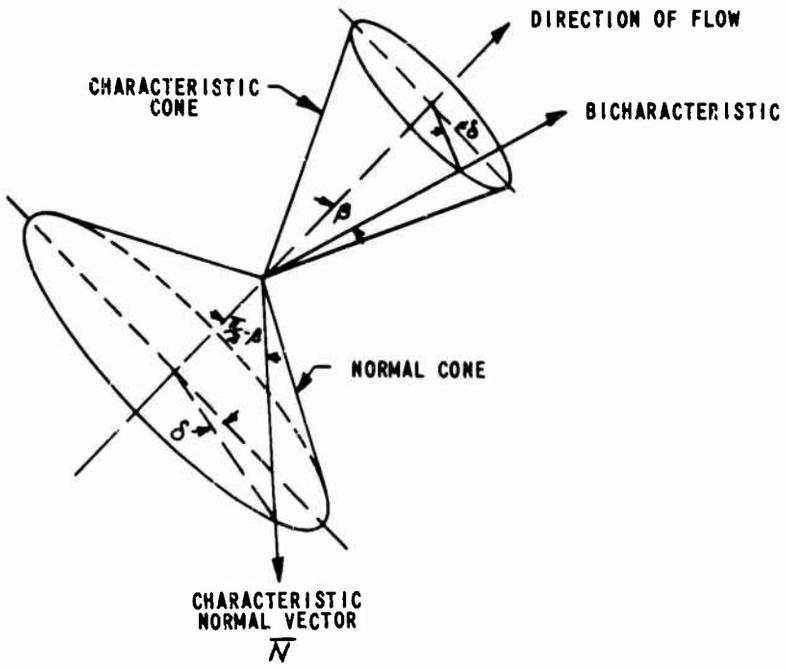


Figure 6 NORMAL CONE AND CHARACTERISTIC CONE

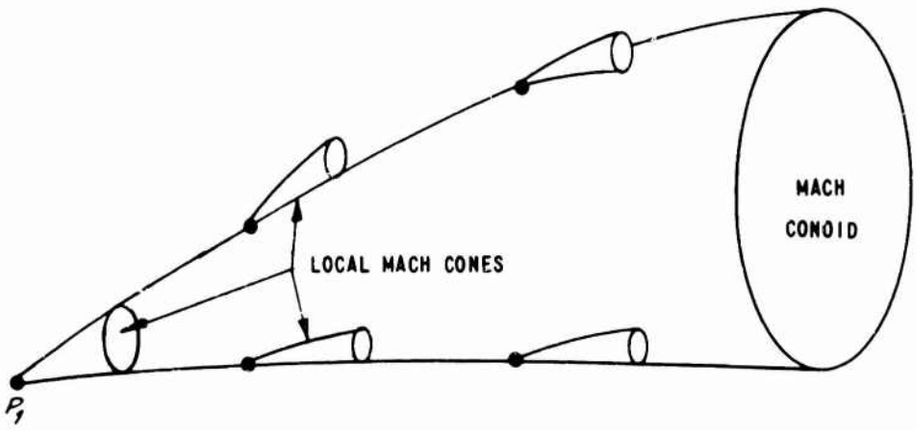


Figure 7 MACH CONOID AS ENVELOPE OF LOCAL MACH CONES

each of three initial points P_1, P_2, and P_3. The lines P_1P_4, P_2P_4 and P_3P_4 then represent numerical approximations to three bicharacteristics and the streamline from point P_4P_5 to the initial plane provides a fourth bicharacteristic direction. The properties at P_5 are determined by interpolation. The equations developed in Section 2 written in finite difference form along these bicharacteristic directions are used to solve for the flow properties at the Point P_4. The main advantage of this network was the fact that the base points P_1, P_2, and P_3 remained fixed throughout the iteration process. However, Sauerwein,[9] in attempting to apply the technique to non-steady flow problems, determined that it was numerically unstable. The reason for this, as shown in Figure 8, is that the domain of dependence of the differential equations (the circle $P_1P_2P_3$) is not contained within the domain of dependence of the difference equations (the triangle $P_1P_2P_3$) and the C-F-L stability condition as introduced in Section 3.1 is continually violated. Sauerwein then proposed what he termed the "modified tetrahedral characteristic line network." In this network (Figure 9), three points P_{12}, P_{23}, and P_{31} representing the points of tangency of the circle inscribed within the original triangle $P_1P_2P_3$ are chosen as new initial points. The properties at points P_{12}, P_{23}, and P_{31} are determined by interpolation and the method then follows the procedure described above for the unmodified network. The C-F-L condition for the points P_{12}, P_{23}, and P_{31} is seen to be satisfied and the technique was found to be stable.

The second method proposed by Thornhill is the "tetrahedral bicharacteristic surface network" (Figure 10). Again, three noncolinear points P_1, P_2, and P_3 are chosen in the initial plane. Point P_4 is determined as the mutual intersection point of the three average internal characteristic planes through lines P_1P_2, P_2P_3 and P_3P_1. The Mach forecone from P_4 defines three bicharacteristic lines P_4P_5, P_4P_6, P_4P_7 as the lines of tangency between the cone and each of the original characteristic planes. The streamline P_4P_8 again furnishes the fourth bicharacteristic direction. The flow properties at the base points P_5, P_6, P_7 and P_8 are determined by interpolation and the flow properties of point P_4 are then solved for along the bicharacteristics as before. This method and the modified method discussed above are essentially the same, as the bicharacteristics chosen run from the tangent points of the circle inscribed in an initial triangle to the new point at which the flow properties are desired.

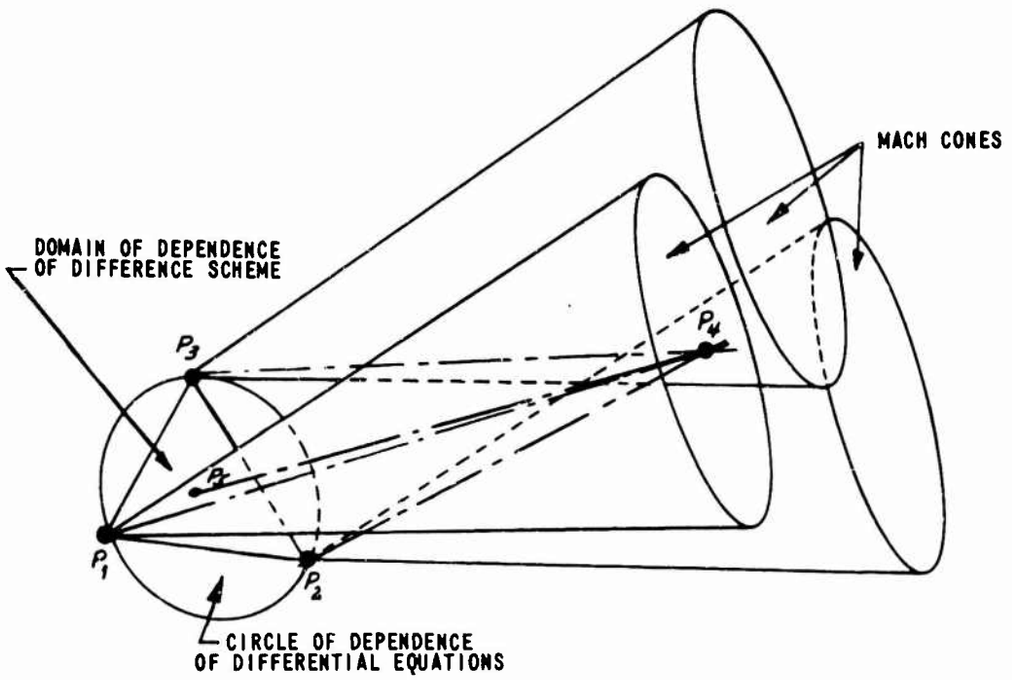


Figure 8 TETRAHEDRAL BICHARACTERISTIC LINE NETWORK

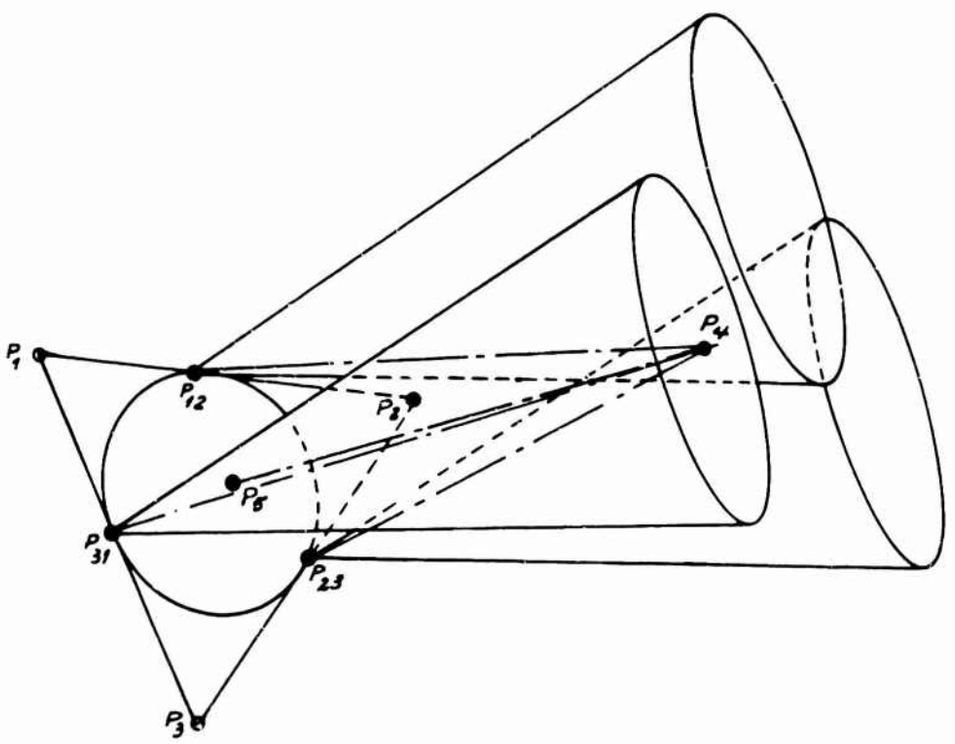


Figure 9 MODIFIED TETRAHEDRAL BICHARACTERISTIC LINE NETWORK

Hence, since the "modified" characteristic network led to a stable solution, it seems plausible to suspect that the present scheme would also be stable. Indeed, such is the case, as was shown by Tsung in applying this technique to the simplified case of three-dimensional potential flow problems.

A "network of intersections of reference planes with characteristic surfaces" (Figure 11) as discussed first by Ferrari,[18] Sauer,[43] and Ferri[44] considers two sets of orthogonal coordinate planes (e.g., y = constant and x = constant) and utilizes a two-dimensional characteristics network. The intersection of the two coordinate planes define lines $P_1 P_2$ and $P_3 P_4$ from which the integration proceeds. Points P_5 and P_6 are located by a forward marching two-dimensional characteristics scheme whereby the characteristics are confined to x = constant planes. Projections of the streamlines through points P_5 and P_6 on to the x = constant planes locate points P_7 and P_8. Lines $P_7 P_5$ and $P_6 P_8$ then serve as two-dimensional streamlines which can be used as auxiliary conditions for determining the flow properties at points P_5 and P_6. The variations of properties in directions across the reference planes x = constant are obtained by using polynomial fits to evaluate the cross-derivative terms which appear in the equations written along the lines $P_1 P_5$ and $P_2 P_5$ or the lines $P_3 P_6$ and and $P_4 P_6$. In order to evaluate the cross-derivatives which occur in the equations, data points must be given in planes z = constant for each step of the calculation. Generally, however, the newly calculated points P_5 and P_6 will not lie in the same z = constant plane. Interpolation or extrapolation to point P_9 in the desired z = constant plane is thus required. While the integration network leading to the new field points in this method does not consist of bicharacteristic curves, Ferrari[45] has shown that the equations used are completely equivalent to the original differential equations and that the existence and uniqueness tests devised by Titt are also satisfied here. Morretti, et al[23] have, in fact, obtained practical results using this scheme. It would appear that the reason Morretti was able to obtain stable results is that the points used as base points for the two-dimensional characteristics plus the points used to determine the cross-derivatives all lie outside the Mach forecone from the new point. Whenever flow conditions are such that the points used do not include the Mach forecone, the procedure would probably be unstable because the C-F-L conditions would be continually violated. Another disadvantage of this network is that interpolation must be

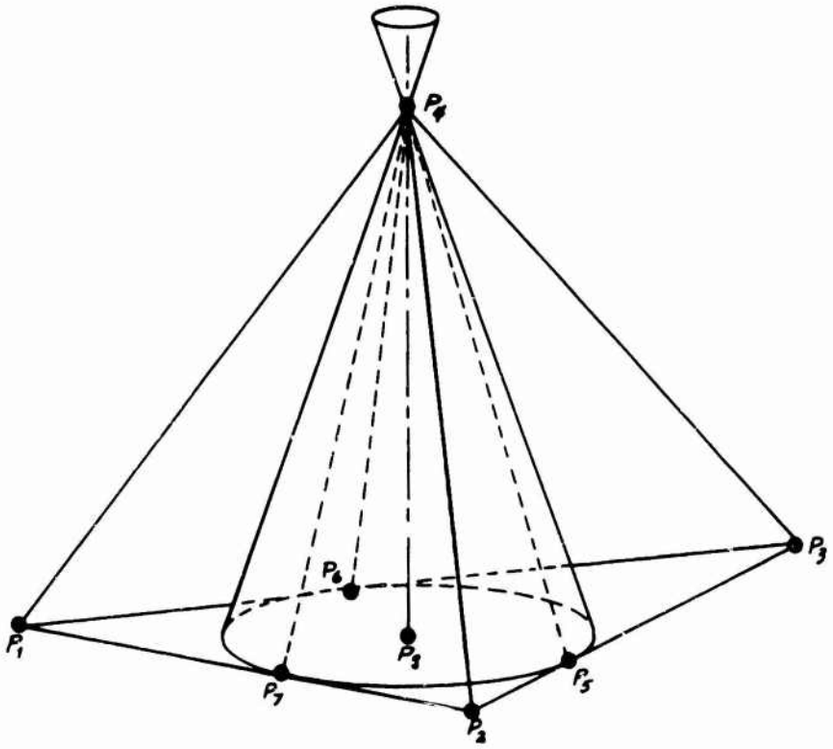


Figure 10 TETRAHEDRAL CHARACTERISTIC SURFACE NETWORK

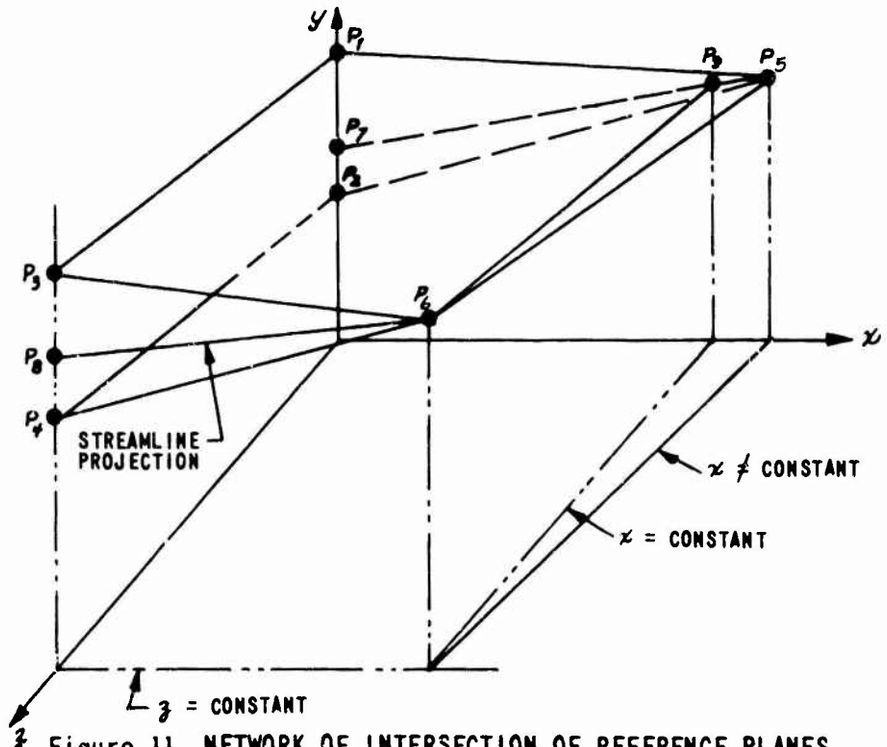


Figure 11 NETWORK OF INTERSECTION OF REFERENCE PLANES WITH CHARACTERISTIC SURFACES

performed throughout the flow field in order to confine the newly calculated points to planes of z = constant. Usually, linear interpolations are used and the over-all effects of these linear interpolations can appreciably affect the accuracy of the technique.

A fifth technique which has had some practical application is that proposed by Butler,[21] (Figure 12). This procedure represents an extension of Hartree's method as described above for the two-independent variable problem. Given the initial data plane, a new point P_5 is chosen in a new data plane. The coordinates of P_5 are thus known exactly. A first estimation of the flow variables at P_5 is made and four bicharacteristics (or one more than the minimum required) located in planes 90° to each other are passed back to the initial surface. Linear combinations of the equations along these four bicharacteristics result in relations involving derivatives in the bicharacteristic directions only at the new point. The streamline $P_6 P_5$ from P_5 provides another relation which is required for obtaining a complete solution. The flow properties at the base points P_1, P_2, P_3, P_4, and P_6 are determined by interpolation. Butler and Talbot have succeeded in applying this technique to some simple two-dimensional unsteady flow problems. A searching scheme was used to determine the nine points which were closest to the particular base point, say P_1, being fit. Then bivariate Lagrange interpolation through these nine points was used to determine the flow properties at P_1. However, the procedure was found to be stable only when all of the base points associated with a given calculation were confined within the same nine-point fit, and hence only when the C-F-L condition was satisfied. If one set of points is used to fit a given base point, while another set of points is used to fit the remaining base points, the procedure was found to be unstable. This can probably be traced to the fact that not all the points in the domain of dependence of the difference equation are treated with equal weight and hence the C-F-L condition is in principle violated.

Other methods have been proposed for which no practical three-dimensional results have as yet been obtained. Holt,[16] using the work of Coburn and Dolph,[10] introduced a network which Fowell called a prismatic network of characteristic surfaces, (see Figure 13). This method, which

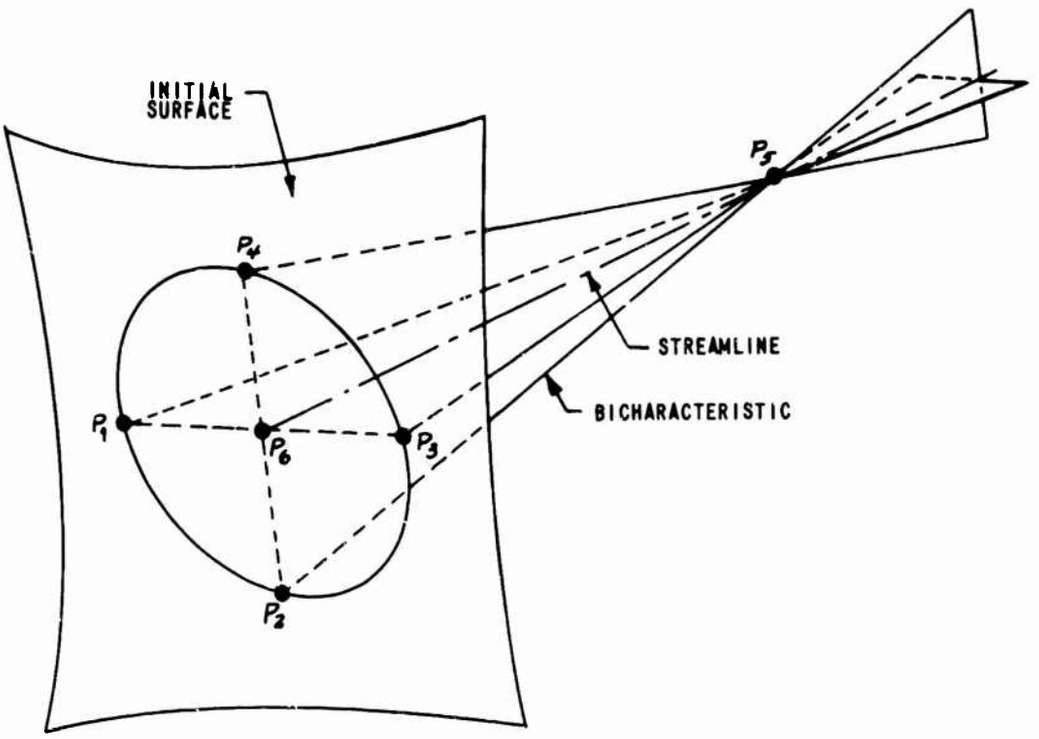


Figure 12 PENTAHEDRAL BICHARACTERISTIC LINE NETWORK

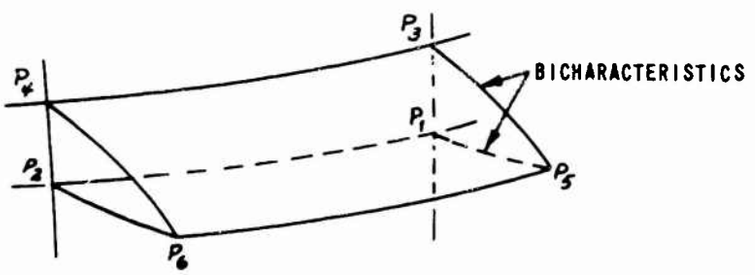


Figure 13 PRISMATIC NETWORK OF CHARACTERISTIC SURFACES

is similar to the network of intersection of reference planes with characteristic surfaces was devised because of the misgivings Coburn and Dolph had regarding the use of non-bicharacteristic directions along which the integrations were to be performed. As mentioned above, however, Ferrari[45] later showed that this objection was clearly unjustified. The method consists of choosing the two bicharacteristic directions P_1P_5 and P_3P_5 as two coordinate directions at a general point. The third direction is provided by the line P_1P_2 in the initial plane. In general, this method requires that the flow properties be known on some surface $P_2P_4P_6$ other than the initial surface (e.g., plane of symmetry, etc.) and this would seem to limit its application.

These are essentially all of the basic characteristic nets which have thus far been advanced. Slight variations of the method of intersection of reference planes and characteristic surfaces have been discussed by Sauer and Shaetz[46] (called by them the method of "near characteristics") for unsteady flow and by Kackova and Cuskin[24] for steady flow. These methods would all seem to have the same advantages and disadvantages of the Ferrari, Sauer, Ferri technique described above.

All of the above integration procedures are seen to require some sort of interpolation at the base of the bicharacteristics in the initial data surface. The "tetrahedral bicharacteristic line network" which sought to avoid this problem was found to be unstable. Also, for every method, interpolations are required at the base of the streamline in the initial plane. A critical point to be considered in a practical application of any of the above techniques then is how to keep the number of interpolations to a minimum. This is necessary both from the point of view of conserving machine time and obtaining accurate representations of flow fields. When nonequilibrium chemistry is included, this point becomes even more salient. Here a total number of $NS-E+NV$ interpolations would be necessary for the compositions and energies at the base of each streamline. However, Curtis[35] has shown that interpolations for species which vary rapidly across the shock layer are a constant source of trouble leading to oscillations in species concentrations along streamlines. If possible, such interpolations should therefore be avoided.

Other considerations should also be taken into account when finite-rate calculations are desired. First, experience has shown that the concentration gradients are usually much steeper along streamlines than are the gradients in other flow quantities. Close control of integration step size along streamlines is thus required in order to obtain reasonably correct concentrations. Second, Sedney[28] and Wood[34] et. al. have shown that the use of entropy as a dependent variable leads to large errors and should be avoided. The energy equation as developed in Paragraph 2.2 has avoided this problem by using temperature as the dependent variable. Third, since any finite-rate chemistry calculations require considerable machine run time per point, a method of reducing the number of points required to obtain an adequate flow field representation is desired. Probably the best way to reduce the number of calculations is to keep the mesh spacing as uniform as possible.

When the integration techniques discussed above are assessed as to their applicability to finite-rate chemistry calculations, the outlook for any of the networks is not very promising. All of the schemes involve interpolations for concentrations in the initial plane. The network of intersections of reference planes with characteristic surfaces would also involve interpolations on the concentrations throughout the flow field. In fact, in order to control mesh size spacings by either adding or dropping points all of the methods would again require interpolations throughout the entire flow field. The methods used to obtain the coordinates of the new point are generally quite complex in all cases except the pentahedral bicharacteristic line network where the location of the new point is initially given. Since the run times involved in finite-rate chemistry calculations are already long, such complicated procedures should in general be avoided.

The integration network described here seeks to avoid the complicated geometry and interpolation problems discussed above. It represents an extension and modification of the two-variable scheme proposed by Curtis[35] and essentially combines the simplicity of the pentahedral network with the advantages of a direct marching technique.

It may be termed a "network of intersections of streamlines with reference planes" since the basic integration coordinate is along the streamline (or particle lines in unsteady flow) and new points are located on successive reference planes or surfaces.

The numerical procedure is illustrated for a single field point calculation in Figure 14. All the flow quantities ---- composition, pressure, flow angle, velocity, etc. are available at initial data points surrounding point P_1, in the non characteristic initial plane $x = x_k$. A new plane at which the data is required is fixed at $x = x_k + \Delta x$. The new point P_2 is first located as the intersection of the streamline from P_1 with the plane $x = x_k + \Delta x$. Four bicharacteristics positioned 90° apart on the Mach forecone from P_2 (Equation 54) are then passed back to the initial plane $x = x_k$. The flow quantities at the base points P_3, P_4, P_5 and P_6 are determined from bivariate surface fitting techniques, remembering that for stability the region fit must include the domain of dependence of the differential equations. Actually, in practice the surface fit for each flow variable is determined only once during the iteration process. Thus, the interpolations become merely a matter of evaluating known polynomial expressions. The velocity q, temperature T, concentrations γ_i and vibrational energies ϵ_j at P_2 are found by integrating Equations (56) through (59) numerically along the streamline P_1P_2. Four compatibility equations (71) written in finite difference form along the four bicharacteristics P_3P_2, P_4P_2, P_5P_2 and P_6P_2 yield appropriate average values of the remaining flow quantities --- pressure p, and flow angles θ and ψ. This procedure is then iterated to convergence of the flow properties at point P_2.

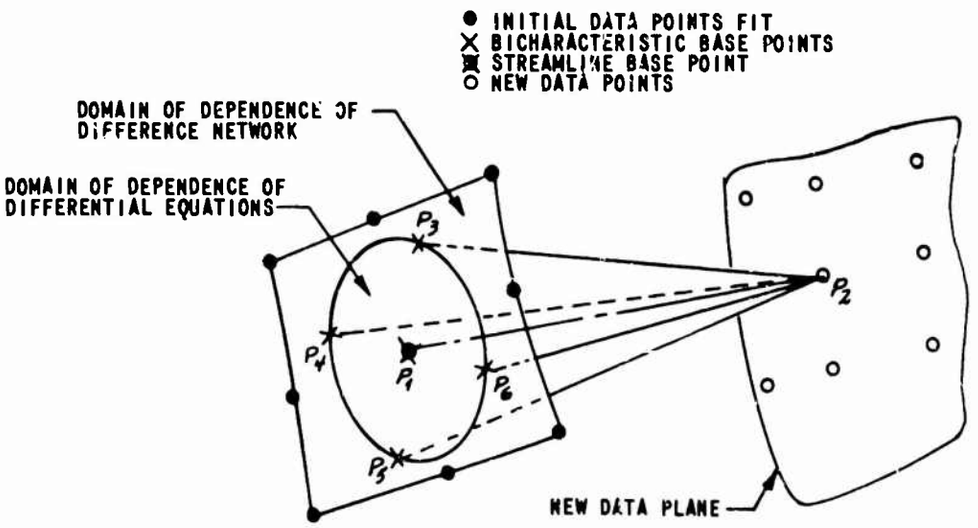


Figure 14 PROPOSED FIELD POINT NETWORK
THREE INDEPENDENT VARIABLES

Perhaps the only disadvantage of "the network of intersections of streamlines and reference planes" is that the base points of the bicharacteristics tend to drift around in the initial plane during the iteration cycle. However, the technique discussed above whereby the surface fitting polynomials are determined first before any calculations are performed would seem to solve this problem. Now the interpolations are relegated to merely solving given polynomial expressions and the extra computational time involved is relatively small. A few less obvious benefits are gained by using this technique. In steady flow, the streamline patterns are easily traced - both along the body and throughout the flow field. The output data is also more easily handled since the flow properties are usually known in planes orthogonal to the body axis, rather than at random nondescript points throughout the flow field.

3.3 PRACTICAL NUMERICAL INTEGRATION PROCEDURE

The preceding considerations illustrate to some extent the network to be used for calculating a point within a flow field. However, in order to calculate an entire flow problem certain boundary points must be included. In general, for the class of problems being discussed here, points that lie on a body surface and shock surface will be required. In the discussion that follows the essentials of the integration process will be included for each of these three basic unit processes. The relationship which exists between the three different types of points and the initial surface is shown in Figure 15.

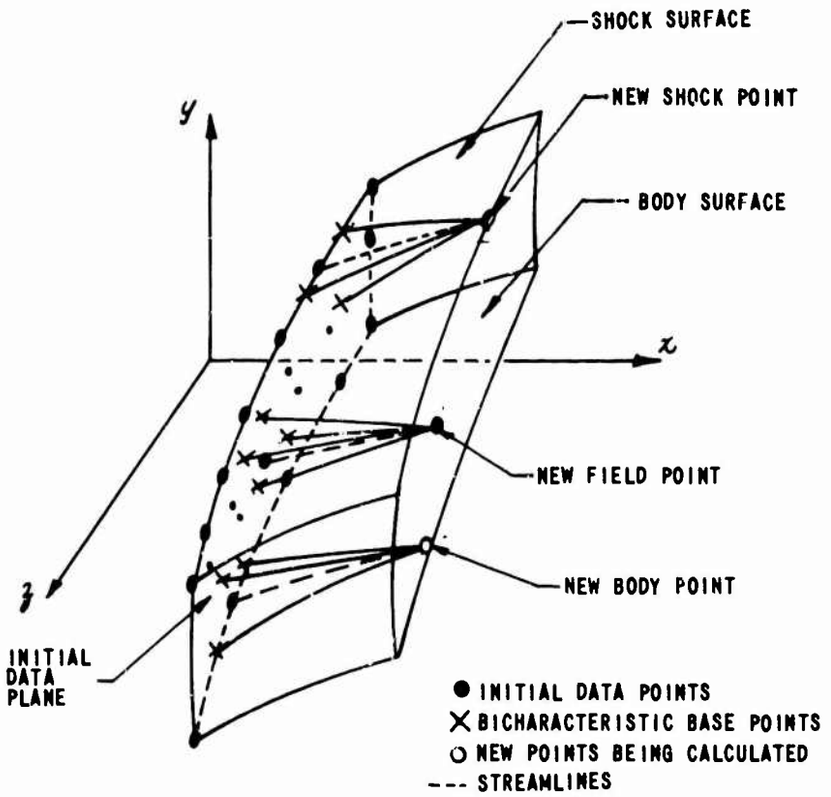


Figure 15 FIELD, BODY AND SHOCK POINTS FOR TWO SUCCESSIVE PLANES

3.3.1 Starting Procedure

The mesh points on the initial, as well as succeeding, data planes are systematized in terms of rays extending out from the body. Each ray on a plane at $x = x_k$ is labeled η and each point on the ray i, so a typical point is given by $P_{i\eta}$. Points $P_{i\eta}$ correspond to body points and points P_{MN} correspond to shock points where $M = \max i$ on ray η. The x, y, z -coordinate system will generally be chosen such that the body axis coincides with the x -axis. A new plane $x = x_k + \Delta x$ is located first. Δx is determined by the C-F-L stability condition which requires that

$$\frac{\Delta \xi}{\Delta \eta} \leq \frac{q}{a} \tag{72}$$

where $\Delta \xi$ is the maximum distance allowed along any streamline, $\Delta \eta$ is usually the minimum mesh spacing and q and a are the minimum velocity and maximum speed of sound respectively. It should be noted that a searching procedure is generally required among the points in the data plane in order that Equation (72) be satisfied. However, this search can generally be confined to regions of the flow field which are a priori known to contain relative maxima and minima of the desired flow variables (planes of symmetry, etc.). Given a point P_1 in the data plane, Equation (72) insures that the Mach forecone from a point P_2 on the streamline through P_1 is contained within the eight initial data points immediately surrounding P_1. (See Figure 14) The nine points shown determine the region in the initial plane over which interpolation surfaces are fitted. If a larger step size $\Delta \xi$ is allowed, a correspondingly larger region in the initial data surface must be fitted (with the resulting larger truncation errors).

3.3.2 Field Point Routine

The solution at a field point involves essentially six separate processes: (1) the surface fits to the flow variables in a region surrounding the initial point are determined, (2) a new field point is located, (3) bicharacteristic lines leading from the new point back to the initial plane are located, (4) the flow quantities at these base points are obtained using the fits developed in step 1, (5) the flow properties at the new point are determined to a first approximation, and (6) the flow properties are iterated to convergence to second-order in the step size.

Step 1

Given an initial Field point P_1 (Figure 14) nine points including P_1 are fit with an exact interpolating polynomial for each function G_i required in the compatibility equation written along the bicharacteristics. For the function G_i it will probably be most convenient to use the pressure p_i, flow angles θ and ψ , Mach angle β , the function $\mathcal{F}$ and the product ρq^2 . In this way computation time can be reduced since the compatibility equation involves only these functions. Compared to the concentration γ_i , each function G_i varies rather slowly throughout the shock layer. As mentioned above, this fact represents a distinct advantage of the proposed scheme.

The coefficients $A1_i$, $A2_i$, $A3_i$, etc. to be used in the interpolation for the function G_i are determined by solving the set of equations

$$(G_i)_j = A1_i + A2_i y_j + A3_i z_j + A4_i y_j^2 + A5_i y_j z_j + A6_i z_j^2 + A7_i z_j y_j^2 + A8_i z_j^2 y_j + A9_i z_j^2 y_j^2 \qquad j = 1,2,\cdots,9 \tag{73}$$

written for the nine initial points $P_j(x, y_j, z_j)$.

Step 2

The new field point P_2 is located as the intersection point of the streamline from P_1 with the given plane $x = k + \Delta x$ as

$$x_2 = x_1 + \Delta x \tag{74}$$

$$y_2 = y_1 + \sin\theta\, dS \tag{75}$$

$$z_2 = z_2 + \cos\theta_1 \sin\psi_1\, dS \tag{76}$$

where

$$\Delta S = \frac{\Delta x}{\cos\theta_1 \cos\psi_1} \tag{77}$$

is the distance along the streamline.

If the new plane is not perpendicular to the x-axis, Equations (74)-(77) along with the equation of the plane locate the coordinates of the new field point.

Step 3

With the new field point P_2 located, the first approximation for the flow properties there are required. Since no calculated properties are available at P_2, they must be estimated initially. A pressure gradient $K = \frac{p_2 - p_1}{p_1}$ is assumed at P_1 while the remaining flow properties at P_2 are taken equal to their corresponding values at the base point P_1.

The compatibility equation (71) corresponding to the Mach forecone from P_2 involves only three dependent variables (p, θ, ψ). It would be reasonable to expect that only three bicharacteristic lines through P_2 would be necessary to solve for these three unknowns. However, while at least three bicharacteristics are indeed necessary, it has been found that more are required in order to give sufficient accuracy to the flow quantities at P_2. The reason for this, as will be discussed in more detail below, is that in writing Equation (71) in finite difference form along the bicharacteristics, certain higher order terms are neglected. Since the iteration scheme to be described in Step 5 below yields values accurate to second order in the step size, in general, these terms are not negligible.

Four bicharacteristics through the new point P_2 are chosen. The base points P_3, P_4, P_5, P_6 in the initial plane are located by solving Equation (54) for four values of the parametric angle $\delta_i = 0, \frac{\pi}{2}, \pi$ and $\frac{3\pi}{2}$. Typically when the initial plane is given by $x = x_k$, point P_i, (i = 3, 4, 5, 6) is located from the following equations,

$$x_i = x_k \tag{78}$$

$$y_i = y_2 - (\sin\theta_2 \cos\beta_2 + \sin\beta_2 \cos\theta_2 \cos\delta_i)\, dL_i \tag{79}$$

$$z_i = z_2 - (\cos\beta_2 \cos\theta_2 \sin\psi_2 - \sin\beta_2 [\sin\psi_2 \sin\theta_2 \cos\delta_i - \cos\psi_2 \sin\delta_i])dL_i \qquad (80)$$

where

$$dL_i = \frac{x_2 - x_i}{\cos\beta_2 \cos\theta_2 \cos\psi_2 - \sin\beta_2 [\sin\theta_2 \cos\psi_2 \cos\delta_i + \sin\psi_2 \sin\delta_i]}, i=3..6 \qquad (81)$$

is the distance along the respective bicharacteristic.

Step 4

The next step is to determine the flow properties at the base points of the bicharacteristics. The flow properties at each base point can be determined by substituting its coordinates into the interpolation formulae derived in Step 1. A new estimate of the parametric angle δ_i is also required at the base of each bicharacteristic. Equations (78) - (81) above with coefficients evaluated at each base point respectively, furnish this new estimate. The actual δ_i used in the computations along each bicharacteristic then represents an average between that at point P_2 and the associated base point.

Step 5

The compatibility equations (56), (57), (58), (59) and (71) are written in finite difference form, and used to find first estimates to the flow quantities at Point P_2. Equation (71) is put in finite difference form by substituting the following approximate expressions for the derivatives of a function f in the bicharacteristic direction.

$$\frac{\partial f_i}{\partial L} = \frac{f_2 - f_i}{\Delta L} \qquad i = 3, 4, 5, 6 \qquad (82)$$

where

$$\Delta L = \left[(x_2 - x_i)^2 + (y_2 - y_i)^2 + (z_2 - z_i)^2\right]^{1/2} \qquad (83)$$

As a result, the following linear equations for the variables p, θ and ψ occur along each bicharacteristic $P_i P_2$ $i = 3, \cdots, 6$

$$\frac{\cot\beta_i}{\rho_i q_i^2}(P_2 - P_i) + \cos\delta_i(\theta_2 - \theta_i) + \cos\theta_i \sin\delta_i(\psi_2 - \psi_i)$$
$$= -\sin\beta_i\left(\cos\delta_i\cos\theta_i\left(\frac{\partial\psi}{\partial N}\right)_i - \sin\delta_i\left(\frac{\partial\theta}{\partial N}\right)_i + \bar{f}_i\right)\Delta L_i \qquad (84)$$

The only unknowns in Equation (84) are the quantities ρ_2, θ_2 and ψ_2 at the new point P_2. One would thus expect that Equation (84) written along three bicharacteristics could be solved simultaneously for the unknowns at P_2. However, before the flow properties at point P_2 can be obtained, the derivatives $\frac{\partial}{\partial N}$ in the non-characteristic direction $\bar{N}$ must be evaluated numerically. The existence of derivatives in the direction normal to a bicharacteristic represents the major difference between two-variable and three-variable method of characteristics. In two-independent variables, the compatibility equations written along the characteristics are ordinary differential equations whose difference analogs can be solved immediately for the flow quantities at the new point. In this sense the compatibility equations are weaker for three-variable problems than the corresponding characteristic equations in two-variable problems. On the other hand, as has been pointed out previously, there are an infinite set of such relations available in the three-variable problem. Butler[21] has utilized some of these extra relations in order to eliminate the derivatives in the non-bicharacteristic direction at the unknown point. Here, extra relations are introduced in order to obtain more accurate average flow properties at the new point. The two techniques should be essentially the same inasmuch as both yield improved approximations to the derivatives in a non-bicharacteristic direction.

Considering any three base points, say P_3, P_4 and P_5, the partial derivatives can be evaluated from

$$\frac{\partial\theta}{\partial N} = \frac{\partial\theta}{\partial x}\frac{\partial x}{\partial N} + \frac{\partial\theta}{\partial y}\frac{\partial y}{\partial N} + \frac{\partial\theta}{\partial z}\frac{\partial z}{\partial N} \qquad (85)$$

where the $\frac{\partial\theta}{\partial x}$, etc. are obtained from equations of the form

$$\theta - \theta_i = \frac{\partial\theta}{\partial x}(x_2 - x_i) + \frac{\partial\theta}{\partial y}(y_2 - y_i) + \frac{\partial\theta}{\partial z}(z_2 - z_i) \qquad (i = 3,4,5) \qquad (86)$$

and the $\frac{\partial x}{\partial N}$, etc. are obtained from the coordinate transformation (68) as

$$\frac{\partial x}{\partial N} = \sin\theta\cos\psi\sin\delta - \cos\delta\sin\psi \tag{87}$$

$$\frac{\partial y}{\partial N} = -\sin\delta\cos\theta \tag{88}$$

$$\frac{\partial z}{\partial N} = \sin\theta\sin\psi\sin\delta + \cos\delta\cos\psi \tag{89}$$

Note that in Equation (86), θ is treated as an unknown so that when (85) is combined with (84) both the right and left hand side of (84) contain the unknowns θ_2 and ψ_2. The procedure for obtaining $\frac{\partial \psi}{\partial N}$ is similar.

With the coefficients thus evaluated, the three linear equations of the form (84) written along the bicharacteristics P_3P_2, P_4P_2 and P_5P_2 can be solved simultaneously to yield first approximations to the flow variables p, θ and ψ (say $p_2^{(1)}$, $\theta_2^{(1)}$, $\psi_2^{(1)}$) at point P_2. Three more bicharacteristics (P_2P_4, P_2P_5 and P_2P_6) are then chosen and using the technique described above another estimate for the flow quantities p, θ and ψ at P_2 ($p_2^{(2)}$, $\theta_2^{(2)}$, $\psi_2^{(2)}$) is determined. Finally, the first approximation to the pressure at P_2 is given by

$$p_2 = \frac{1}{2}\left(p_2^{(1)} + p_2^{(2)}\right) \tag{90}$$

Similarly, the average values of θ_2 and ψ_2 are formed. It should be noted here that more sets of bicharacteristics can be located and still more first estimates to the flow properties at P_2 determined. Then Equation (90) would represent an average over all such estimates. In general, however, if four bicharacteristics are positioned evenly around the Mach forecone from P_2, the addition of any other estimates at P_2 would probably exceed the overall accuracy of the remainder of the scheme and the final results would be changed only slightly.

The compatibility equations (58) - (59) along the streamline P_1P_2 are integrated to yield the remaining flow quantities at point P_2. In general for nonequilibrium flow, the temperature [illegible], concentrations [illegible] and vibrational energies ε_j, vary much more rapidly from point P_1

to point P_2 than do the pressure p, and flow angles θ and ψ. Hence, the equations along the streamline cannot usually be replaced by their first differences and integrated directly as was done above for the bicharacteristic equations. Following Wood et. al.[34] and Curtis[35], the pressure is assumed to vary linearly along the streamline P_1P_2 and a fourth-order Runge-Kutta technique is used to integrate the given coupled ordinary differential equations in several steps along P_1P_2. Although the Runge-Kutta method is generally satisfactory, for many cases of high-temperature air flows (e.g. conditions where both fast and slow chemical rates are present in the same region) it fails to prevent instabilities from arising. Treanor has recently developed an algorithm designed especially for handling these so-called "stiff" equations which often occur in reacting and relaxing flows. The method which is similar to the Runge-Kutta technique includes terms which are fifth order in the integration step size. The details are given in Reference 47 and will not be repeated here. However, it should be emphasized that the calculation time per integration step is generally quite long because several rather complicated derivatives must be evaluated.

Step 6

Having determined a first approximation to the flow properties at a new field point the last step is to utilize an iteration process which determines the position of the new point and the flow properties there. The iteration process employed here is similar to the "modified Euler method" or "Huen's first method"[48] whereby steps 2 through 5 are repeated using average values of the flow quantities. Along the streamline the coefficients involving θ_1 and ψ_1 are replaced by average values of the following form

$$\sin\theta_{1,2}^{(n)} = \frac{1}{2}\left(\sin\theta_1 + \sin\theta_2^{(n-1)}\right) \tag{91}$$

wherever they occur in Equations (74), (75), (76) and (77). Here, the superscript refers to the current iteration cycle. Similarly, in Equation (78) through (81) the coefficients defining the bicharacteristic directions are supplanted by appropriate average values. As an example, Equation (79) now becomes

$$y_{i+2}^{(n)} = \frac{1}{2}\left(\sin\theta_2^{(n-1)}\cos\beta_2^{(n-1)} + \sin\beta_2^{(n-1)}\cos\theta_2^{(n-1)}\cos\delta_{oi}\right.$$

$$\left. + \sin\theta_i^{(n-1)}\cos\beta_i^{(n-1)} + \sin\beta_i^{(n-1)}\cos\theta_i^{(n-1)}\cos\delta_i^{(n-1)}\right) dL_i^{(n)} \qquad (92)$$

$$(i = 3,4,5,6)$$

The coefficients of $(p_2 - p_i)$, $(\theta_2 - \theta_i)$, $(\psi_2 - \psi_i)$ and ΔL_i in the compatibility Equation (84) are also replaced by the average of their values at P_2 and $P_i^{(n)}$ respectively.

The iteration cycles may be continued until $p_2^{(n)} = p_2^{(n-1)}$, $\theta_2^{(n)} = \theta_2^{(n-1)}$, $\psi_2^{(n)} = \psi_2^{(n-1)}$, etc. within the prescribed limits. Usually it is only necessary to test the convergence on the pressure p, since experience here and in the two variable problems have indicated that whenever p has converged the remaining flow quantities have converged also.

Using the modified Euler integration scheme described above along with the averaging technique presented in Step 4, the truncation error is third-order in the step size; that is, the procedure is accurate to second order in the step size as is the case with two-variable characteristics. When the averaging discussed in Step 4 is not included, the process is probably limited to first-order in the step size through the assumption that the derivatives $\frac{\partial\theta}{\partial x}$, $\frac{\partial\theta}{\partial y}$ and $\frac{\partial\theta}{\partial z}$ as given by Equation (85) are constant over the network. Thus, the addition of an extra bicharacteristic relation is seen to be essential in order to achieve the desired accuracy.

3.3.3 Body Point Routine

Calculation of a point on the body surface involves essentially the same six steps which were necessary for a successful field point routine. The body surface will be assumed to be given by

$$B(x, y, z) = 0 \tag{93}$$

B represents either a known function or a surface fitting element in the region of the body point being calculated. In this way, rather general body shapes and motions can be accounted for. The geometry of a typical body point calculation is shown in Figure 16.

Step 1

The first step of the body point routine is exactly the same as that for the field point procedure. Nine initial points including eight nearby the initial body point P_1, are fit with interpolating surface elements of the form of Equation (73) for each of the variables ρ , θ , ψ , β , γ and ρq^2 . Note, from Figure 16, that of the eight points chosen, only two are body points, the remaining six being field points.

Step 2

The coordinates of the new body point are obtained as the intersection of a plane through P_1 formed by the body unit normal (n_1 , n_2 , n_3) and the unit tangent velocity ($\frac{u}{q}$, $\frac{v}{q}$, $\frac{w}{q}$) vector at P_1 with the body surface (1) and the given plane $x_2 = x_1 + \Delta x$. Since the normal plane is

$$(\bar{N} \times \bar{S}) \cdot d\bar{L} = \begin{vmatrix} \underset{i}{n_1 \cos\theta \cos\psi} & \underset{j}{n_2 \sin\theta} & \underset{k}{n_3 \cos\theta \sin\psi} \end{vmatrix} \cdot d\bar{L} \tag{94}$$

This leads to simultaneous solution of the equations

$$B(x, y, z) = 0 \tag{95}$$

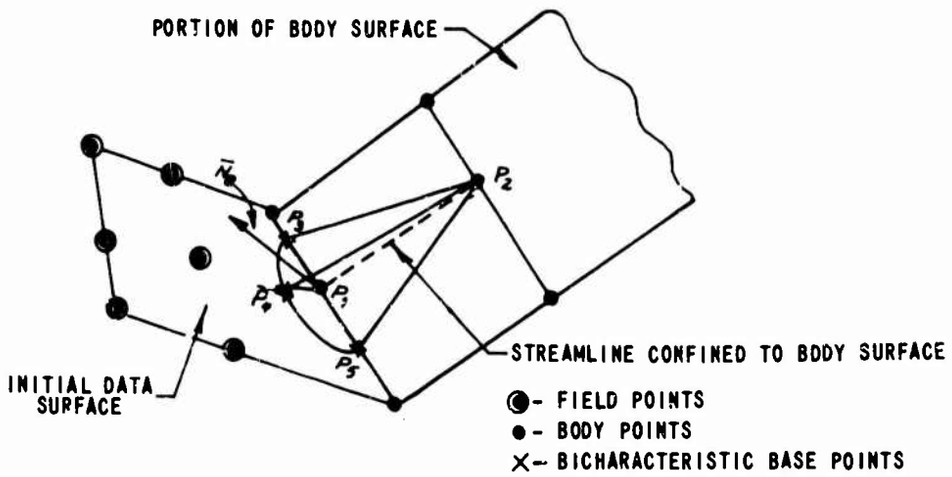


Figure 16 BODY POINT NETWORK - THREE INDEPENDENT VARIABLES

and

$$\begin{aligned}&\left[(n_3)_1 \cos\theta_1 \cos\psi_1 - (n_1)_1 \cos\theta_1 \sin\psi_1\right](y - y_1)\\ &+ \left[(n_1)_1 \sin\theta_1 - (n_2)_1 \cos\theta_1 \cos\psi_1\right](z - z_1)\\ &= + \left[(n_3)_1 \sin\theta_1 - (n_2)_1 \cos\theta_1 \sin\psi_1\right](x_2 - x_1)\end{aligned} \tag{96}$$

The solution defines the body point P_2 (x_2, y_2, z_2). Equations (95) and (96) can be solved by using the second order Newton-Raphson algorithm with $y^{(0)} = y_1$, and $z^{(0)} = z_1$, serving as initial guesses to the coordinates of point P_2.

Step 3

As in the case of the field point network, most flow quantities at the new body point P_2 are initially taken equal to their respective values at point P_1. The initial guess for pressure will again be given by $P_2 = P_1 (K+1)$ where K is a known gradient.

Bicharacteristic lines leading from point P_2 back to the initial plane are determined next. Three bicharacteristics will generally be required here in order to obtain sufficient accuracy in the calculations at P_2. Two bicharacteristic elements are located in the tangent plane to the body surface at P_2 and the third in the normal plane containing the streamline P_1P_2. Equations (78) - (81) serve to define the base points P_i (i = 3, 4, 5) when δ_i at point P_2 is known. Since the streamline P_1P_2 lies on the body surface, the parametric angle δ_1, defining the bicharacteristic in the normal direction, is given by the scalar product of the unit normal vector (n_1, n_2, n_3) and the coordinate vector $\bar{\eta}$ (Equation 90) shown in Figure 1.

$$\delta_3 = \cos^{-1}\left(-(n_1)_1 \sin\theta_1 \cos\psi_1 + (n_2)_1 \cos\theta_1 - (n_3)_1 \sin\theta_1 \sin\psi_1\right) \tag{97}$$

Then,

$$\delta_{4,5} = \delta_3 \pm \frac{\pi}{2} \tag{98}$$

define the remaining two bicharacteristic directions through the new body point P_2.

Step 4

The flow quantities at the base points are determined from the interpolation surface fits found in Step 1. This step is exactly the same as Step 4 in the field point procedure.

Step 5

The next step is the determination of the flow properties at point P_2. For the body point, this involves the solution of two compatibility equations together with the boundary condition of flow tangency at P_2. Choosing two of the three bicharacteristics located in Step 3 (say P_3P_2 and P_4P_2), the derivatives $\frac{\partial}{\partial N}$ which appear in the compatibility equations along each bicharacteristic must be computed. These derivatives are evaluated in much the same manner as that used for the field point solution. Now, however, in evaluating $\frac{\partial \theta}{\partial x}$ etc. in Equation (86) the initial body point P_1 is used as one of the base points.

The compatibility equation (84) along P_3P_2 and P_4P_2 are two equations in three unknowns p_2, θ_2 and ψ_2. The pressure p_2 is eliminated from each to yield one equation relating θ_2 and ψ_2. The conditions of flow tangency provides the additional relation between θ_2 and ψ_2 as

$$(n_1)_2 \cos\theta_2 \cos\psi_2 + (n_2)_2 \sin\theta_2 + (n_3)_2 \cos\theta_2 \sin\psi_2 = 0 \tag{99}$$

Using a bivariate Newton-Raphson procedure these two equations can be solved for initial estimates to the flow angles at P_2. The pressure is then found from one of the original compatibility equations. Repeating the above for bicharacteristics P_3P_2 and P_5P_2 another estimate for θ, ψ and p at P_2 is obtained. The final first approximation to the pressure and flow angles at P_2 is represented by the average of these two solutions.

The remaining flow properties at P_2 are obtained by integrating the compatibility equations (57) through (59) along the streamline in the same manner as was used in the field point routine.

Step 6

Using the "modified Euler process" described in Step 6 of the field point procedure, Steps 2 through 5 are iterated to convergence. The location and properties of point P_2 are then stored and the computation continued to the next point. This completes the body point solution for steady flow.

3.3.4 Shock Point Routine

The shock point process is the most complex of the three basic unit processes required, although it follows the same basic pattern as has already been established. Now, however, not only must the new shock point be located and the composition and flow properties there calculated, but the orientation of the shock surface at the new point must also be determined. The geometry involved in the shock point solution is shown in Figure 17.

Step 1

This step is exactly the same as Step 1 of the body point procedure with the body points being replaced by three shock points. The form of surface fit chosen and the flow variables fitted are the same as those given in Step 1 of the field point procedure.

Step 2

Since, unlike the body, the location of the shock is not known, a point on the shock will be located as the point of intersection of a line tangent to the shock surface through the initial shock point P_1 and the plane $x = x_1 + \Delta x$. The tangent line lies in the plane formed by the shock normal vector $\bar{N}_{s_1}$, and the velocity vector $\bar{q}_1$. Its direction is, therefore, chosen as that of the velocity component tangent to the shock surface from the point P_1 and is given by

$$\bar{r}_1 = \bar{N}_{s_1} \times \frac{\bar{q}_1 \times \bar{N}_{s_1}}{|\bar{q}_1 \times \bar{N}_{s_1}|} = r_{x_1}\bar{i} + r_{y_1}\bar{j} + r_{z_1}\bar{k} \qquad (100)$$

With $\bar{r}_1$ established, the position of the new shock point is

$$x_2 = x_1 + \Delta x \qquad (101)$$

$$y_2 = y_1 + \frac{(r_y)_1}{(r_x)_1}\Delta x \qquad (102)$$

$$z_2 = z_1 + \frac{(r_z)_1}{(r_x)_1}\Delta x \qquad (103)$$

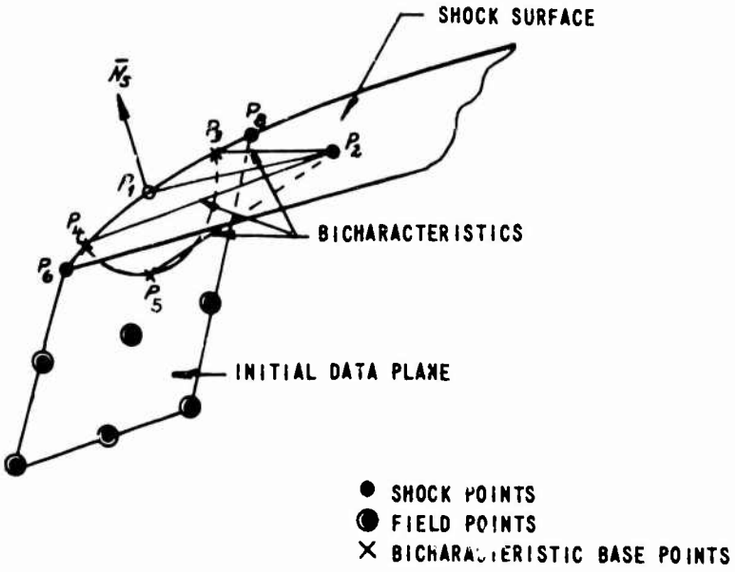


Figure 17 SHOCK POINT NETWORK - THREE INDEPENDENT VARIABLES

Step 3

The flow conditions behind the shock at the new shock point P_2 are determined by solving the shock wave equations (Rankine-Hugoniot equations) using given values of the normal $\bar{N}_{S_2}$ and the "free stream" conditions at P_2. The normal $\bar{N}_{S_2}$ is initially chosen as

$$\bar{N}_{S_2} = \bar{N}_{S_1} + \bar{K}_1 \Delta x \tag{104}$$

where $\bar{K}_1$ is a known gradient based upon the two previous shock points. ($\bar{K}_i = \frac{\partial \bar{N}_{S_i}}{\partial x_i}$) The flow properties incident to the shock $\bar{q}_{2_+}$, p_{2_+}, ρ_{2_+}, $(\epsilon_{j_2})_+$ and (γ_{i_2}) are either known or they can be determined by interpolation between the closest known points in the incident stream. (In this way, both internal and external shock wave surfaces may be handled by using exactly the same technique.) Quantities ahead of and behind the shock are denoted by a plus or minus sign, respectively. Letting the subscripts N and T denote the normal and tangential velocity components, respectively, the shock equations can be written in the form

$$\rho_+ q_{N_+} = \rho_- q_{N_-} \tag{105}$$

$$p_+ + \rho_+ q_{N_+}^2 = p_- + \rho_- q_{N_-}^2 \tag{106}$$

$$h_+ + \frac{1}{2} q_{N_+}^2 = h_- + \frac{1}{2} q_{N_-}^2 \tag{107}$$

$$q_{N+} = q_+ \sin\sigma \tag{108}$$

$$q_{T+} = q_+ \cos\sigma \tag{109}$$

$$q_{T_+} = q_{T_-} \tag{110}$$

$$\gamma_{i+} = \gamma_{i-} \tag{111}$$

$$\epsilon_{j+} = \epsilon_{j-} \tag{112}$$

where σ is the shock wave angle.

In the general case, these equations must be solved numerically for conditions behind the shock. To this end, given the shock wave angle σ at the point P_2, the conditions across the shock can be found using the iterative method of Garr and Marrone[39]. Here, the pressure p_{2-} is initially assumed and equations (105) - (107) above are solved for the remaining unknowns ρ_{2-}, $(q_{N_2})_-$ and h_{2-}. The process is then iterated by varying the pressure until the total enthalpy behind the shock matches that ahead of the shock within a specified relative error. The flow velocity components are next calculated from the known values of $\bar{N}_{s_2}$ and $\bar{q}_{N-}$ as

$$u_{2-} = u_{2+} + N_{s_x}(q_{N-} - q_{N+}) \tag{113}$$

$$v_{2-} = v_{2+} + N_{s_y}(q_{N-} - q_{N+}) \tag{114}$$

$$w_{2-} = w_{2+} + N_{s_z}(q_{N-} - q_{N+}) \tag{115}$$

Then, the flow angles $(\theta_2)_-$ and $(\psi_2)_-$ are given by

$$(\theta_2)_- = \tan^{-1}\left(\frac{v_2}{\sqrt{u^2+v^2}}\right) \tag{116}$$

$$(\psi_2)_- = \tan^{-1}\left(\frac{w_2}{u_2}\right) \tag{117}$$

Thus, all flow properties behind the shock wave are known at the new shock point P_2.

Step 4

Next, the locations of three bicharacteristics leading from the new shock point P_2 back to the initial plane $x = x_k$ are desired. In a similar manner to that used in the body point solution, two of the bicharacteristics are selected to lie in the shock surface, while the third is chosen to intersect the initial surface at a point in the field. The base points P_3 and P_4 on the shock surface in the initial plane are determined as the points of intersection of the Mach cone (Equation (48) with

the shock surface and the initial plane $x = x_k$. To this end, the element $P_6P_1P_8$ (see Figure 17) of the shock surface in the initial plane is fitted with a quadratic in y and z as

$$y = Az^2 + Bz + D \tag{118}$$

Then, equations (118) and (48) yield the coordinates of the two base points P_3 and P_4. The parametric angles δ_i defining the bicharacteristics P_3P_2 and P_4P_2 can be obtained from Equation (50) with coefficients evaluated at P_2 and

$$\Delta L_i = \left((x_2 - x_i)^2 + (y_2 - y_i)^2 + (z_2 - z_i)^2\right)^{1/2} \quad i = 3,4 \tag{119}$$

Then, a third bicharacteristic is located midway between P_3P_2 and P_4P_2 on the Mach forecone by taking

$$\delta_2 = \frac{\delta_1 + \delta_2}{2} \tag{120}$$

at point P_2 and solving Equation (54) for the coordinates y_5, z_5.

Step 5

This step is exactly the same as Step 4 in the field and body point procedures. Given the coordinates of base points P_3, P_4 and P_5 the function surface fits evaluated in Step 1 can be used to determine the base point flow properties.

Step 6

In Step 3, all flow quantities behind the shock at point P_2 were determined by solving the Rankine-Hugoniot equations. These flow properties must also satisfy the compatibility equations written along the bicharacteristics P_3P_2, P_4P_2 and P_5P_2 when the shock wave is correctly located and oriented at P_2. This fact can be used to develop an iteration scheme to converge the position and orientation of the shock wave element through P_2.

Since conditions behind the shock at any point cannot be multi-valued, whenever one flow property is correct the remaining properties must be right also. Consider that the flow angles $(\theta_2)_-$ and $(\psi_2)_-$ as determined in Step 3 above, are correct. Then, each of the three compatibility equations written along the bicharacteristics P_3P_2, P_4P_2 and P_5P_2 can be solved to yield three values of the pressure $p_2^{(i)}$ at P_2. The partial derivatives $\frac{\partial}{\partial N}$ which occur in the compatibility equations are evaluated here in the same manner as was used in the field and body point procedures. (Equations (85) - (89).)

Comparing the average pressure $p_{2_-}^*$ as obtained from the compatibility equations with the pressure p_{2_-} from the shock relations, a new estimate to the shock normal $\bar{N}_s$ at P_2 can be determined by defining

$$e^{(n)} = p_{2_-}^{(n)} - p_{2_-}^{*(n)} \tag{121}$$

Then, new values of the direction cosines of the normal $\bar{N}_{s_2}$ are selected by linear interpolation as

$$N_{s_{x_2}}^{(n+1)} = \left(N_{s_x}^{(n-1)} e^{(n)} - N_{s_x}^{(n)} e^{(n-1)}\right) \Big/ \left(e^{(n)} - e^{(n-1)}\right) \tag{122}$$

and

$$N_{s_{y_2}}^{(n+1)} = \left(N_{s_y}^{(n-1)} e^{(n)} - N_{s_y}^{(n)} e^{(n-1)}\right) \Big/ \left(e^{(n)} - e^{(n-1)}\right) \tag{123}$$

while,

$$N_{s_{g_2}}^{(n+1)} = \left(1.0 - N_{s_x}^{\,2} - N_{s_y}^{\,2}\right)^{1/2} \tag{124}$$

Again, the superscript (n) denotes the iteration cycle. This procedure is repeated in an iteration process using average flow properties and coefficients to locate the new shock point and the bicharacteristic base points as in the field and body networks. Average values are also used for the coefficients of the respective compatibility equations associated with the bicharacteristics P_3P_2, P_4P_2 and P_5P_2. Iteration continues until

$$\left| \frac{p_{2-} - p_{2-}^{*}}{p_{2-}} \right| \leq \text{Test Constant} \qquad (125)$$

at which time the position and orientation of the shock wave are found that yield flow properties satisfying the compatibility equations to the desired degree of accuracy. This completes the shock point solution.

3.4 SIMPLIFIED FLOW MODELS

The methods derived above apply to three dimensional steady gas flows in which chemical reactions and vibrational relaxations occur at finite rates. Experience with finite rate programs[35] indicates that even for calculations involving two independent variables the full equations are apt to require long machine times - especially in near equilibrium flow situations. Therefore, it may not always be desirable or even necessary to consider the full nonequilibrium flow equations. Instead, following Curtis[49], appropriate simplifications of the chemistry may be introduced. Specifically, three sets of simplifying assumptions relative to the chemistry and the thermodynamics of the gas flow are presented here. Each one is listed below, along with the procedures required for carrying it out.

3.4.1 Ideal Gas Model

In this case, all chemical production terms are equated to zero and the specific heat C_p is constant corresponding to no excitation of the vibrational degrees of freedom. Therefore, the variables and terms involving them may be eliminated from all preceding equations: NS, NV, γ_j, ϵ_j, R_{v_j}, Q_i and $\bar{F}$. The speed of sound is computed from

$$a = \sqrt{\frac{\Gamma T}{\Lambda}} \tag{126}$$

and for steady flow the velocity is given by

$$\frac{1}{2} q^2 \Lambda = (H_\infty - C_p T) \tag{127}$$

3.4.2 Frozen - Inhomogeneous Model

The frozen-inhomogeneous model corresponds to the case where the gas composition along streamlines (or particle paths) has frozen in a partially dissociated state. The composition does vary across streamlines however, and thus throughout the shock layer the gaseous mixture is inhomogeneous. Assuming vibrational equilibrium, frozen flows can be calculated by setting the following terms equal to zero in the compatibility equations: NV, R_{v_j}, Q_i, ϵ_j and $\bar{F}$. The frozen specific heat C_p is obtained from Equation (28) with $NV = 0$ and the frozen speed of sound is again given by (34). The netword of intersections of streamlines and reference planes is seen to be suitable for calculating frozen flows because the concentrations of the various species are forced to remain constant along each streamline.

3.4.3 Equilibrium Model

The equilibrium model assumes that local thermochemical equilibrium will be established at every point in the flow field. Again, R_{v_j}, ϵ_j, NV, Q_i, γ_i and $\bar{F}$ are set equal to zero in the compatibility equations. Two distinct approaches are possible here. First, the temperature, density and speed of sound could be determined from polynomial fits[50] or special curve fits[23] of tabulated air data.

In this case, the equilibrium rather than the frozen speed of sound is used. Knowing the pressure and temperature at each point in the field the equilibrium concentrations can be determined uniquely. A second technique would be to use an iterative method to solve the full coupled equilibrium equations at every point in a manner similar to that used by Boyer[51]. While the latter scheme is more accurate, the first method would involve far less computational time and should probably be used in any multi-dimensional characteristics program. The accuracy of the approximate method can generally be held within 1% which is sufficient for almost all practical purposes.

3.5 INITIAL VALUE SURFACE

The body point, field point and shock point solutions described above can be used as building blocks from which the complete flow field can be constructed. However, before the method of characteristics can be utilized to solve such flows, an initial value surface on which the flow properties are completely specified must be provided. In many cases, obtaining these starting values is not a simple problem. In the discussion that follows several possible procedures which provide the required starting regions for rather specific problems are discussed.

In the case of a completely general three-dimensional body, a plane of inputs along which the flow is initially supersonic is required. Unfortunately, since the necessary data will in general be asymmetric, a method for obtaining inputs in this general case does not currently exist. Consequently, more specialized solutions must be attempted. One such specialization consists in solving the flow field in the vicinity of a spherically-capped body at angle-of-attack. In this case, initial data in a plane normal to the flow direction is axisymmetric and may be provided by using any of the well-known axisymmetric-blunt body solutions. Among these, either the direct method using integral relations[52] or an iterated inverse method[39] would yield initial data which is sufficiently accurate for continuing the solution using the method of characteristics. The only restriction to such an approach is that for the angles of attack considered, the body sonic point on the compression side of the body must be located on the spherical nose.

Certain three-dimensional, steady blunt body flows may also be solved by treating them as limiting solutions in time of a properly posed unsteady flow problem. These methods, which have recently been advanced in the literature[9, 53], consist first of moving from a known or attainable steady flow solution to another desired steady solution using a prescribed body motion. The body motion may consist of either a systematic warping from an initial shape (e.g., a spherical cap) to a final shape (e.g., an ellipsoidal nose) or of moving a given body from one configuration (e.g., flow at zero angle of attack) to another (e.g., flow at high angle of attack) or any combinations of these two. The final steady flow is obtained by continuing the calculations in time with the desired body surface unchanged until the flow transients due to the prescribed motions have disappeared. Since as has been pointed out before, the governing equations for unsteady flow are hyperbolic, the method of characteristics for unsteady flow may be conveniently used as a basis for solving the steady flow problem using this approach. Here, starting with a known flow field in the initial hyperplane $t = t_o$, (e.g., the steady flow solution about a three-dimensional nose) a series of unsteady three-dimensional flow problems are solved step-by-step in time until the flow field no longer changes with time. Although these computations may indeed be very tedious and involve long computer run times, the flow fields about some rather arbitrary bodies can be determined.

Section 4
RESULTS AND CONCLUSIONS

4.1 PROBLEM AREAS AND THEIR SOLUTIONS

Before presenting the results obtained from the three-dimensional steady flow method of characteristics program, it would seem to be desirable to first discuss some of the difficulties which arose during the check out and debugging phases of the program development. This discussion should prove helpful to those who attempt to utilize and/or modify the basic method as it was presented in Section 3. The overall problems were stability, accuracy, and computer running time, in that order of importance. These problems did, however, prove to be interrelated, since once greater accuracy was attained the computer run time improved considerably.

4.1.1 Interpolation

As discussed previously, when the integrations are performed by tracing streamlines step-by-step through the flow field the points in each plane normal to the x-axis do not maintain a uniform mesh spacing. Consequently, what was required was a bivariate interpolation procedure which was applicable to a net of irregularly distributed points. Unfortunately, because of the generality of the problem, very little attention beyond that of just stating the difficulties involved, [54, 55] has been given to it in past work. Generally, using polynomials, only two basic approaches to this problem are deemed useable. The first possibility would be to use a method of least squares approximation whereby the number of base points through which the fit is to be passed exceeds the number of coefficients which are required in the fitting polynomial. Using this procedure the sum of the squares of the errors represented by the differences between the value of a given function at a point and the value obtained for that function from the fit should be a minimum. The other possibility is to use an exact polynomial fit through the given base points. By "exact" here is meant that the error between the actual value of the function and the values obtained from the fit is zero at least to within the accuracy of the machine computations. The form of each of these fits depends in turn upon the relative orientations of the base points, the number of base points chosen and the form of the basic independent set of

basis functions which comprise the fit. For example, for linear fit in one variable, x, the basis functions would be f and x).

Both of the above methods were investigated for use with the three-dimensional characteristics program. While these studies were were not exhaustive, a few definite conclusions seemed to be justified. First, for bodies with "rounded" cross-sections (i.e., axisymmetric or elliptical cross-sections which are not too flat), as would be expected, better fits could be obtained by using combinations of the polar coordinates (r, ω) as the basis functions for the fits rather than the cartesian coordinates (y, z). This was especially true in the case of the flow angles where relatively large changes occur around the body. For bodies which have locally two-dimensional regions, it would appear that probably a better set of basis functions would be combinations of the cartesian coordinates (x, y). This point has not yet been thoroughly checked out however, and probably even for these areas the polar fits will be satisfactory. If not, the program can readily be changed to allow for using different basis functions over different regions of the flow field. In the case of an elliptical cross-section this could best be accomplished by translating the center of the (y, z) coordinate system from the center of the ellipse to the focus, and then using the (r, ω) - fit over the region of the field covering the "rounded" portion of the ellipse and the (y, z) - fit over the remaining part.

The accuracy of each surface fit was improved considerably by scaling the independent variables with respect to the coordinates of the central point used in the fit. In many cases, improvements in accuracy amounting to two significant figures were obtained by utilizing this technique. Finally, for the ranges of points tested (e.g., nets of 9 and 25 points centered around a base point in the initial plane) with the polar fits, the method using least squares and that using the exact fit yielded approximately the same accuracies for all functions fit. However, since the exact fit involves fewer operations and does in fact fit the data at the given points exactly, this method was finally chosen for use with the program. The justification for using an exact fit lies also in the fact that the data at each

known point is itself in a sense "exact" for a given problem and hence, smoothing procedures using least squares or other methods should not be required. However, as was indicated above, these results are somewhat preliminary and now that the characteristics program is working, a more thorough investigation of the surface fitting procedures, as applied to several different flow fields, should be undertaken.

4.1.2 Flow Angle Derivatives

A more fundamental difficulty occurred while attempting to apply the characteristics procedure to compute some simple two-dimensional and conical flow fields. Two flow problems were studied initially, (1) a simple wedge and (2) a sharp cone - both at zero angle of attack. In each case, the minimum number of bicharacteristics (three) necessary to determine the flow properties at a new field point was utilized. The wedge flow was successfully reproduced to eight places in all flow variables during each axial step. This, however, was not a very demanding test because the flow field is initially uniform and no iteration was required in order to determine the flow properties at a new point. Also, the normal derivatives in the compatibility equations were in this case identically zero. The flow over a 10° right circular cone offered a more stringent test on the overall calculation procedure. Although the initial data was axisymmetric to eight places, the results in this case were non-axisymmetric. The degree of non-axial symmetry varied according to the gradients existing in the flow and the step size. However, the pressure and flow angles were generally constant to only three significant figures (1 to 2%) around the cone after just one or two axial steps. These azimuthal variations in the flow properties around an axi-symmetric body could be attributed to the fact that as the calculations progressed around a given ring of data points (r = constant), the relative orientations of the three bicharacteristics which were used to determine the flow properties with respect to the azimuthal plane through the initial data point were different. Thus, the conditions at each new point would be expected to be slightly different, in that conditions at the respective base points, as determined from the interpolations, were not the same. The variations which

did in fact occur were surprisingly large. These large variations were attributed to the fact that some first order effects were included in the compatibility equations through the assumption that the derivatives of the flow angles with respect to the coordinate directions were constant. Hence, in order ot obtain meaningful results, better approximations to the derivatives were required. These were obtained by introducing extra bicharacteristics at each point and averaging the flow properties at the new point in the manner set forth in Section 3. Actually, in the program three sets of bicharacteristics (initial set is rotated 40°) are utilized to obtain this average. The averaging of the flow properties at the new point improved the accuracy of the calculation by as many as two signifigant figures. In addition, the more accurate proceedure converged in three cycles rather than the seven required without averaging; thus, although more calculations were required within each cycle, the net run time was not affected by the averaging procedure.

4.1.3 Mach Cone Intersects the Shock

Another difficulty was encountered when a new ring of field points was added near the shock wave as a result of introducing new streamlines from the previous shock point. In this case, a portion of the Mach forecone from a new field point intersects the shock wave surface. Thus, the base points of some of the bicharacteristics, which are used to calculate the flow properties at the new field point, intersect the initial data plane on the free-stream side of the shock surface. Two different methods were utilized in attempting to solve this problem. First, the relevant portion of the shock surface and the flow quantities behind the shock were fitted by surface fits similar to those used in interpolating for the base points. These fits were used to determine the coordinates of and flow properties at the base points of the bicharacteristics which intersected the shock surface, while those bicharacteristics on the portion of the Mach cone which did not intersect the shock were handled in the usual manner. Even though this procedure seemed to work properly, the results obtained were not compatible with the accuracy of the solution in the

remainder of the flow field. For an axisymmetric flow field, the results along the new ring were axisymmetric to only three places, while the rest of the flow field remained so to six significant figures. The reason for the discrepancy is thought to be due to the fact that the base points associated with each of the two separate fitting schemes are treated with unequal weights since not the same number of bicharacteristics intersect the shock as intersect the field. Time limitations prevented further development of this method. Instead, another much simpler technique was employed. In this scheme, the C-Γ-L stability criterion was in principle violated by allowing some of the bicharacteristics to pass through the shock surface and intersect the initial plane outside of the shock layer. The flow properties at these base points are determined by extrapolation, utilizing the interpolating surface fits which had already been set up surrounding the initial shock point. This procedure yielded results which were compatible with the remainder of the flow field solution, and is the scheme that is presently being used with the program. Note, that violating the stability criterion in a small region does not seem to cause course instabilities to arise. This is because basically, instability is the result of a cumulative growth of errors throughout a calculation and does not usually result from a few isolated errors, as is the case here. In order to prevent the extrapolations from being extended too far outside the shock wave, the step size is halved before any points in the new plane are determined, whenever a new ring of field points is to be added. Note that since the streamline is inclined toward the shock wave, only a small portion of the Mach forecone from a new point on the streamline will intersect the shock when the step size ΔX, is small. Thus the extrapolation described should be reasonably accurate as the results obtained thus far using this technique seem to indicate.

4.1.4 Problems Associated with Large Angles of Attack

In general, as discussed above, orienting the X-axis (the axis along which the calculations proceed) along the free stream wind direction has the advantage that the initial input data is axisymmetric and can be

obtained from two-independent variable programs. However, at large angles of attack (say 30°) the flow in the shock layer on the windward side is at a relatively low Mach number. Consequently, the Mach lines are steeply inclined and the step size Δx between successive data planes is small. In fact, whenever the sum of the local two-dimensional flow angle χ and the Mach angle Δx is greater than 90° (see Figure 18) the step size Δx becomes negative and the calculations would proceed upstream. The program is presently incapable of handling this situation. It should be emphasized however, that this limitation does not represent a deficiency in the overall method, but is rather a problem due to the necessity of aligning the x -direction with the free stream-velocity vector in order to utilize axisymmetric inputs. Whenever a method of obtaining non-axially symmetric inputs becomes available the orientation of the x -axis can be changed so that the condition $\theta + \chi = 90°$ is less likely to occur.

When the axial step is positive but small due to the low Mach number of the flow in the shock layer, calculations of complete flow fields would be speeded up considerably by reorienting the x -axis parallel to either the body axis or to the windward side of the body. This idea was investigated by rotating successive data planes by a small angular increment $\Delta\mu$ about a point on the windward shock (Figure 18), followed by a coordinate transformation to a body-axis system. Unfortunately, programming difficulties and lack of time forestalled the realization of this scheme.

4.2 TYPICAL OPERATIONAL EXPERIENCE WITH THE PROGRAM

The results of typical field, body and shock point calculations are given in Appendix B, Tables B-I, B-II, and B-III. The number of iterations shown are typical for calculations to date with the field and body generally requiring three to five cycles depending upon the gradients and the axial step size, while the shock requires at least five. Convergence at a shock point is slower, because the first two cycles involve first, an estimate of the direction of the shock normal, and then a given small change in this direction in order to numerically determine the derivatives in the Newton-Raphson

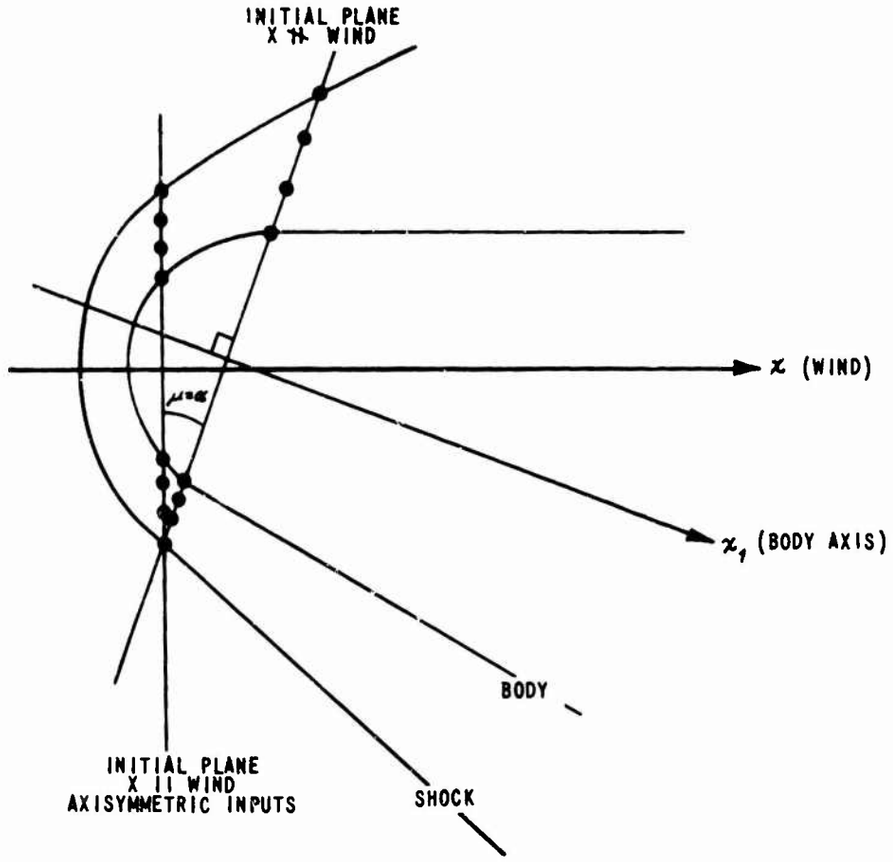


Figure 18 COORDINATE SYSTEM ROTATION

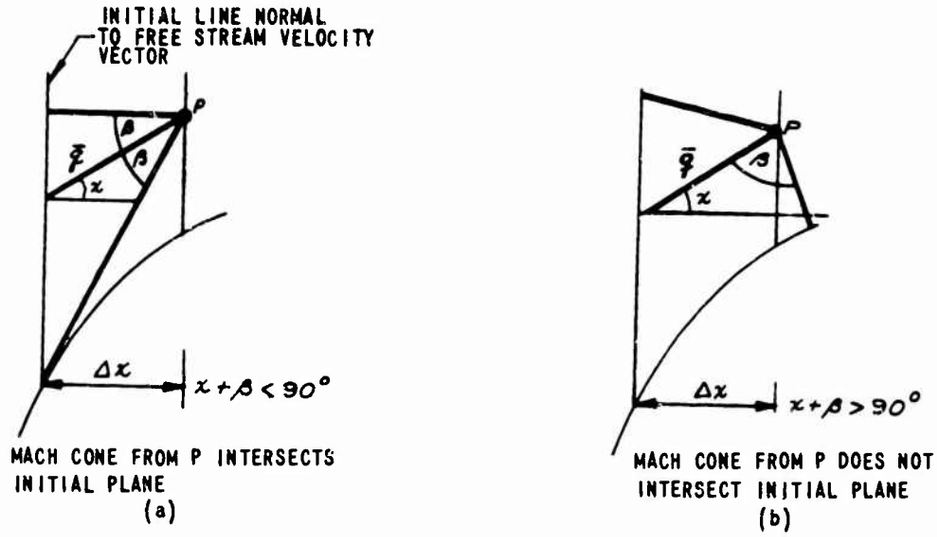


Figure 19 ILLUSTRATION OF DIFFICULTIES ENCOUNTERED IF FLOW MACH NUMBER IS TOO LOW

procedure used to obtain the next values of shock parameters. Thus, the actual internal iteration process does not begin at the shock point until the third cycle of the overall iteration has been reached. It then usually takes three more cycles to complete the shock point computation. The results indicate that the coordinates and flow angles generally converge faster than do the pressure and velocity, which usually converge at the same rate. Actually, just the pressure has been used for the convergence tests thus far in the program and from the results obtained this seems to work quite well; however, in some cases, it might be better to test on both the velocity magnitude and the pressure - especially if a smaller convergence criterion than 10^{-5} is used. It should be noted that although the results in Tables B-I through B-III are given in terms of eight digit numbers corresponding to the number of digits the machine carries, only the first four or five digits are significant, due to the various truncations which occur during the computation.

The total computation time required to completely solve for the coordinates and flow properties at a particular field, body or shock point is difficult to determine accurately. It depends upon the gradients which exist in the flow, the spacing between data points, the axial step size and the convergence test criterion utilized. An estimation of the time required can be obtained by dividing the total time taken to calculate several planes by the total number of points calculated. Although this time includes the time required to read input data from the tape, set up the initial data surface, and write data back on output tape as desired, the significant portion consists of the actual computation time - especially if several planes (1000 to 2000 points) are computed. Using this technique, the time ranged from 0.30 to 0.50 seconds per point on the IBM 7094 computer. The comparable time for the IBM 7044 would be about twice this value. Thus, it is absolutely necessary to decrease this time significantly, if many problems are going to be run using the present method. This will be especially true when the method is extended to include nonequilibrium thermochemical effects.

4.3 RESULTS FOR COMPLETE FLOW FIELDS

Three different afterbody flow cases were computed using the three-dimensional method of characteristics program. These consisted of the flow fields associated with a spherically blunted 15° cone at both 0° and 10° angles-of-attack, and a spherically capped elliptical afterbody whose cross-sections were ellipses with the eccentricity of each varying as a function of the axial distance x at 0° angle-of-attack. Since the flows around the cone at 0° angle-of-attack and the elliptical body are symmetric with respect to the y and z axes, it is sufficient in these cases to calculate only one quadrant of the entire field. In the case of the blunted cone at angle-of-attack however, there exists only one plane of symmetry (the plane containing the free stream velocity vector and the y-axis) and a full 180° of the field must be computed.

All computations were performed using the same input conditions obtained in the manner discussed in Paragraph 4.2. The radius r_o, of the nose sphere in each case was 0.28 cm with the center of the reference (x, y, z) coordinate system located at the center of the sphere. An initial line of nine data points, equally spaced between the body and the shock, was provided by the two independent variable characteristics program at a non-dimensionalized (with respect to r_o) axial coordinate x equal to -0.4534. This initial data line was parallel to the y-axis in the xy-plane. To provide an initial plane of data points the initial line was rotated about the x-axis in given increments $\Delta\omega$ of the polar angle ω. A different number of meridional rays were used for each of the problems studied. The free-stream conditions were in all cases: $M_\infty = 8.33$, $\rho_\infty = 3.153 \times 10^{-7}$, $T_\infty = 250°K$ corresponding to an altitude of 200,000 feet. Results for each case studied will be presented separately.

4.3.1 Blunted Cone At Zero Angle-of-Attack

Calculation of the flow field about a blunted cone at zero angle-of-attack was carried out using five initial meridional rays located 22.5° apart. It took approximately 205 steps to reach a distance of 8 nose radii down the body. The computer run time was nearly 90 minutes on the IBM 7094 computer.

While this run time is long, it is far from the optimum that could be obtained using the program even in its present state of development. Since limited computer time was available for each of the flow calculations described here, it was decided to allow the machine to choose the axial step size based upon the minimum mesh spacing in the field so that stability would be assured, and the program would proceed as far as possible without the necessity of re-running a problem. While this objective was accomplished, the step sizes the program chose were generally much smaller than would be necessary for stability. This was because the minimum mesh spacing generally occurs between the last point on the ray and the point on the shock surface. When new streamlines are added at the shock, this mesh size is generally an order of magnitude smaller than the spacings between the rest of the points in the field. Consequently, the step size used after adding points near the shock is an order of magnitude smaller than the maximum that would probably be used in such computation. It usually takes five or six axial steps for the step size to again approach a reasonable value. A possible solution to this would be to set the axial step size at a given value based upon the C-F-L condition, and then calculate several steps using this constant value. Then the step size could be changed again and another few steps could be computed, and so on. If the procedure were followed here, the results indicated that an average step size somewhere between 0.07 and 0.08 $\left(\frac{\Delta x'}{r_o}\right)$ could be used. This improvement alone, would cut the calculation time by a factor of approximately two. (A short test case indicated this to be feasible).

Another way the computation time could be decreased would be to use fewer streamlines throughout the flow field. In the present computation, streamlines were added at the shock on an average of once every 10 steps, so that a total of 21 points were specified along each ray in the final plane. This is probably more than enough to adequately describe the flow field for most ideal gas problems. By dropping selected streamlines (e.g., every other one), computing time would be reduced because one needs to calculate fewer points, and in addition the axial step sizes can be increased as a direct consequence of the larger spacings between data points. Between 10 and 20 points on each ray should be adequate to solve steady flow problems for an ideal gas.

The shock wave shape and some typical streamlines are shown in Figure 20. Since the scales used along the abscissa and ordinate are different, the shapes of these curves are somewhat distorted; however, the body streamline does correspond to a 15° cone. As can be seen, the shock shape agrees very well with that obtained from the two-independent variable characteristics solution,[49] the maximum variation amounting to about 0.7% at x = 7.5. The body shape and streamline patterns are also as would be expected. Note the manner in which the program adds streamlines along the shock, as the calculations proceed in the axial direction. The addition of these extra streamlines is seen to be absolutely necessary if sufficient accuracy is to be maintained throughout the flow field.

Results for the flow quantities at the shock and body in one meridional plane are presented in Figures 21 through 24. In all cases, the data remained axially-symmetric to at least four significant figures over the entire range of the computation. Figure 21 shows a comparison between the pressures obtained from the present program, and the two variable program. The results are in good agreement and indicate that the truncation errors are approximately the same for both methods (order of the step size squared). The pressure along the body reaches a minimum at approximately $\chi = 4.25$ and then recompresses toward a constant sharp cone value as is expected. The total variation in pressure across the shock layer is nearly 8% greater using the three variable characteristics scheme than with the two variable method at an axial location $\chi = 7.5$. However, this discrepancy seems to be decreasing and as cone flow is reached, it should nearly disappear. The velocity V 2-D flow angle θ, and temperature T are shown in Figures 22, 23 and 24 respectively, compared to the expected axisymmetric results. These comparisons again clearly show the accuracy of the technique with the maximum variations amounting to less than 1.5%.

Note in Figures 21 and 22 the appearance of a rather severe expansion for the 3-D solution on the body at the point of tangency between the spherical nose and the conical afterbody ($\chi = -0.24$) compared to the axisymmetric value. The over-expansion is approximately 10% of the expected value, and is propagated throughout the flow field in the form of

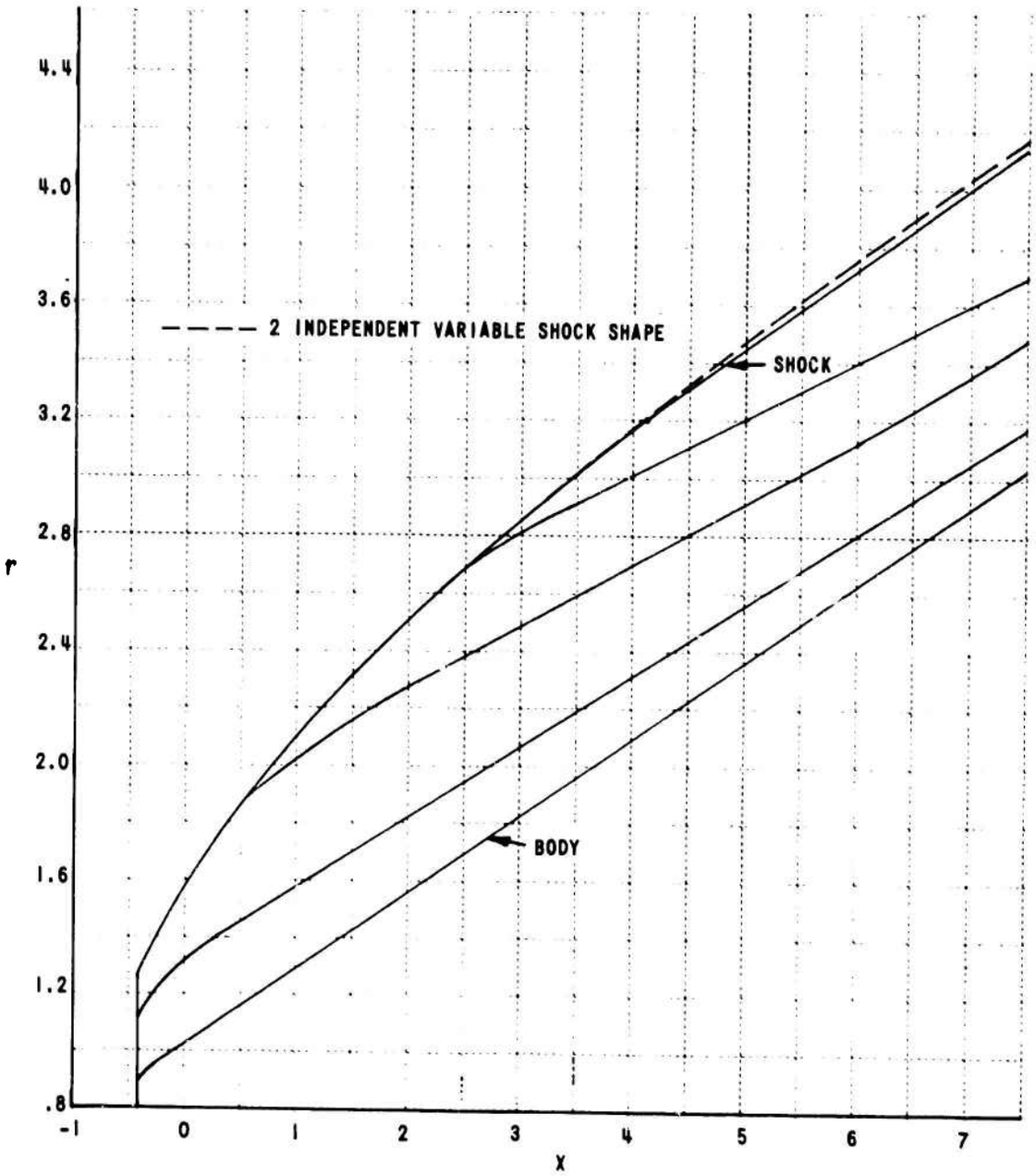


Figure 20 TYPICAL STREAMLINE SHAPES FOR BLUNT 15° CONE, $\alpha = 0°$

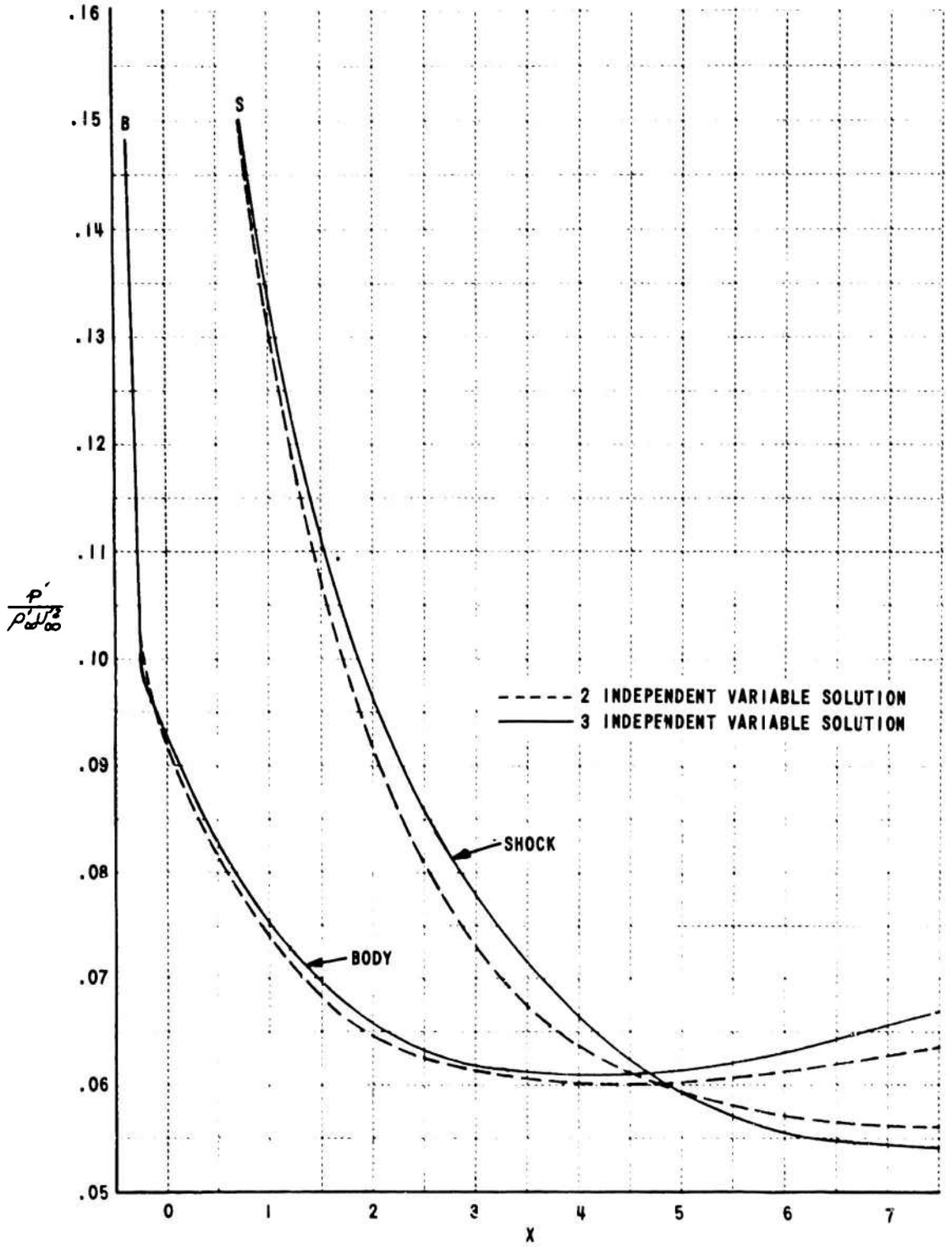


Figure 21 PRESSURE DISTRIBUTION ON BLUNT 15° CONE, α = 0°

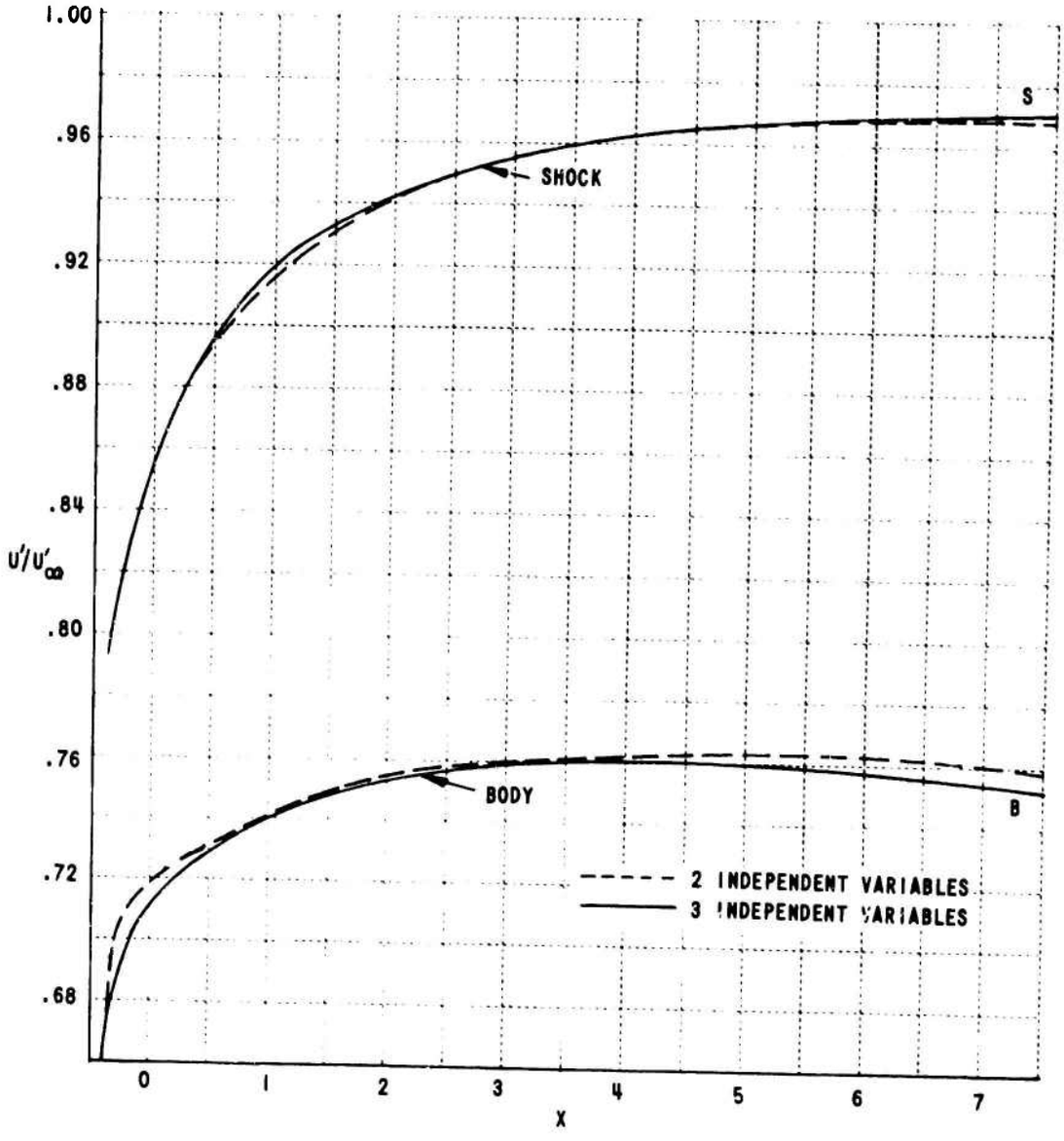


Figure 22 VELOCITY DISTRIBUTION ON BLUNT 15° CONE, α = 0°

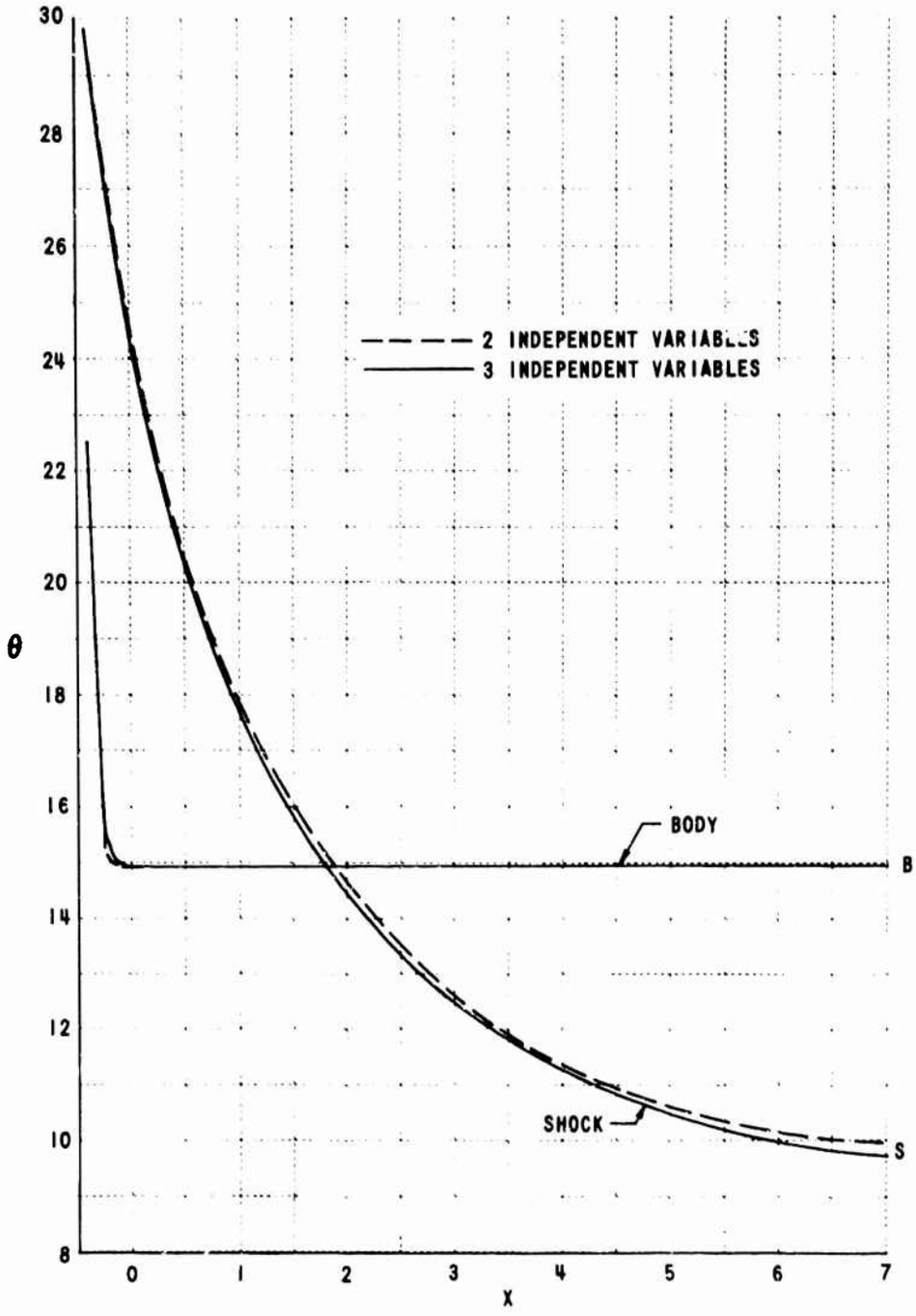


Figure 23 FLOW ANGLE, θ, FOR BLUNT 15° CONE, α = 0°

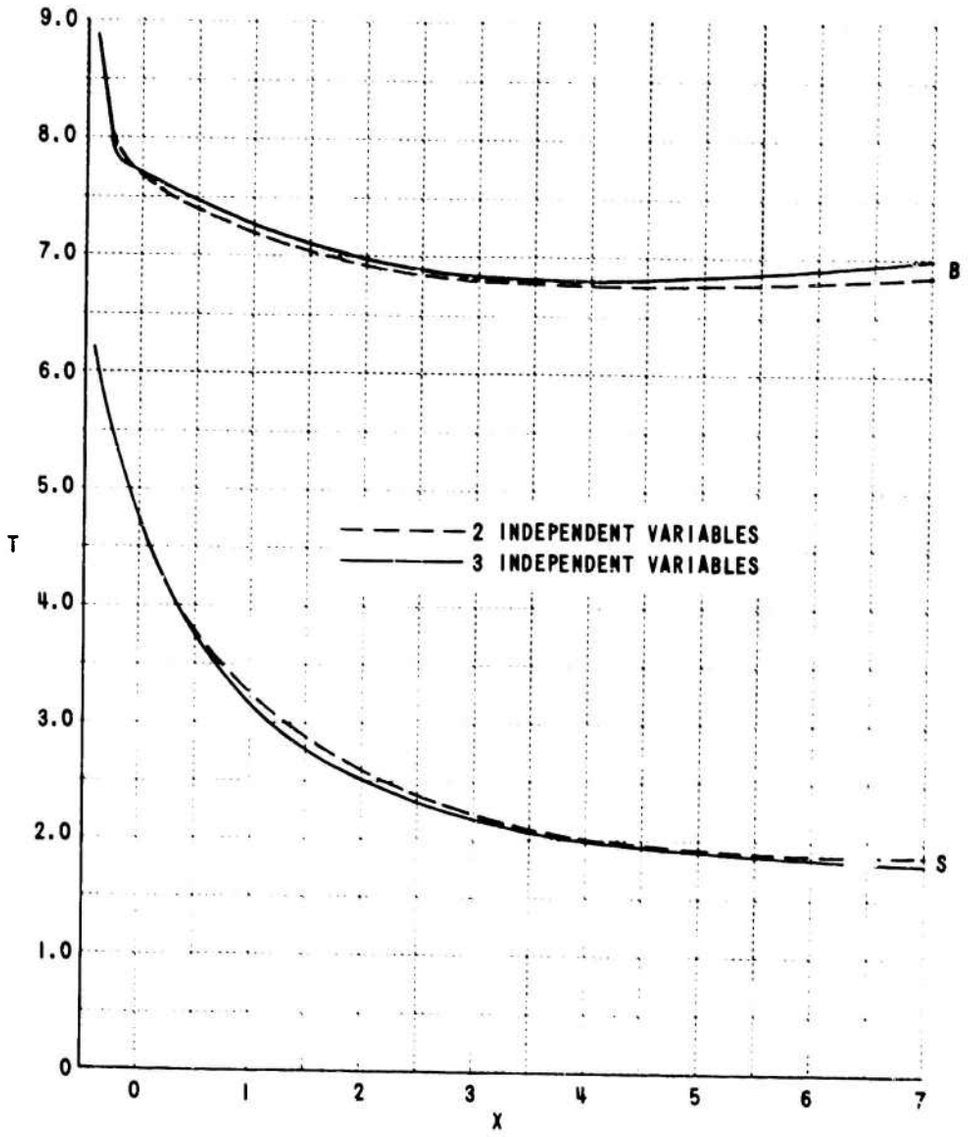


Figure 24 TEMPERATURE DISTRIBUTION FOR BLUNT 15° CONE, $\alpha = 0°$

an expansion wave. Figures 25, 26 and 27 show the propagation of this wave along rays normal to the body axis for three different axial locations. As shown in Figure 30 the expansion has just begun to form near the body at an axial location of $\chi = -0.231$ and the pressure throughout most of the shock layer corresponds to the expected axisymmetric value.

4.3.2 Blunted Cone at 10° Angle-of-Attack

The flow field about the same 15° blunted cone was computed at 10° angle-of-attack using initially 16 meridional rays located 11.33° apart. Because of computing time limitations, the computations were halted after an χ coordinate equal to 3.1 nose radii was reached. At the time the computation was stopped, it was proceeding smoothly and could have been continued, if desired. The calculations were proceeding at the rate of approximately one nose radius per hour on the IBM 7094. However, the same remarks concerning step size and mesh spacing that were given above for the zero angle-of-attack case, would also apply here. As an example, the average step size, $\Delta\chi$, used in the calculation, was about $\Delta\chi = 0.03$ (nondimensionalized with respect to nose radius) while a short run indicated that step sizes as high as $\Delta\chi = 0.07$ could be used without causing instability for at least 20 axial steps. It appears that optimizing the step size, perhaps using the method discussed previously, is required in order to improve the efficiency of the calculation.

Another way the running times could be reduced would be to use the body axis as the χ- axis for the computation, rather than a wind-oriented axis (See Figure 19) as was utilized here. The shift to the new axis could be accomplished by rotating the x-axis to coincide with the body axis. This would have the effect of decreasing the maximum flow angles and as a result increase the effective step size.

The technique of dropping selected "rings" of data points was successfully tried in this calculation. When 10 extra rings had been added, for a total of 19, every other ring throughout the field was dropped from the computation. The step size increased by a factor of 2 as would be

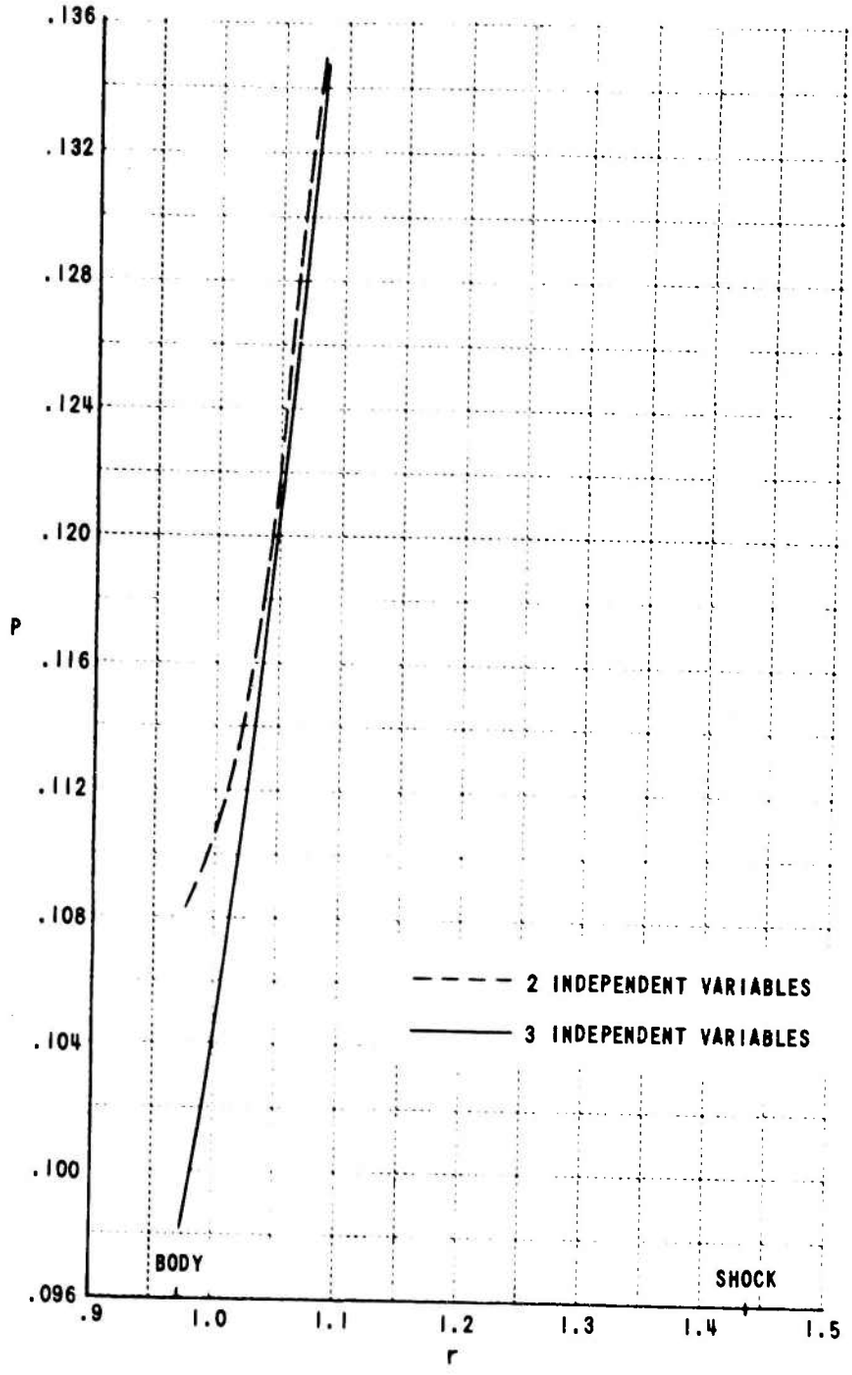


Figure 25 PRESSURE DISTRIBUTION THROUGH SHOCK LAYER, BLUNT 15° CONE, α = 0°, X = -0.231

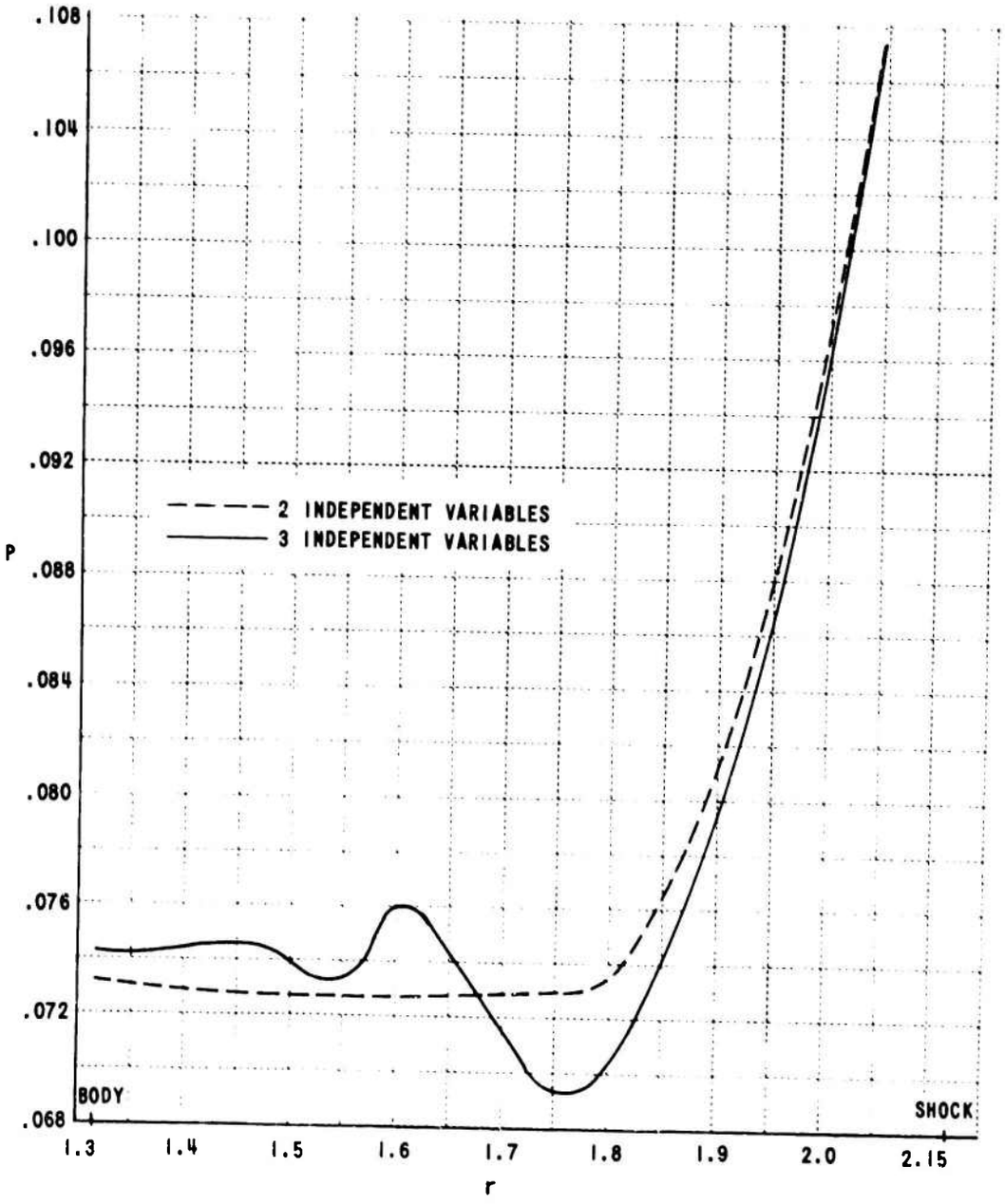


Figure 26 PRESSURE DISTRIBUTION THROUGH SHOCK LAYER, BLUNT 15° CONE, α = 0°, X = 1.058

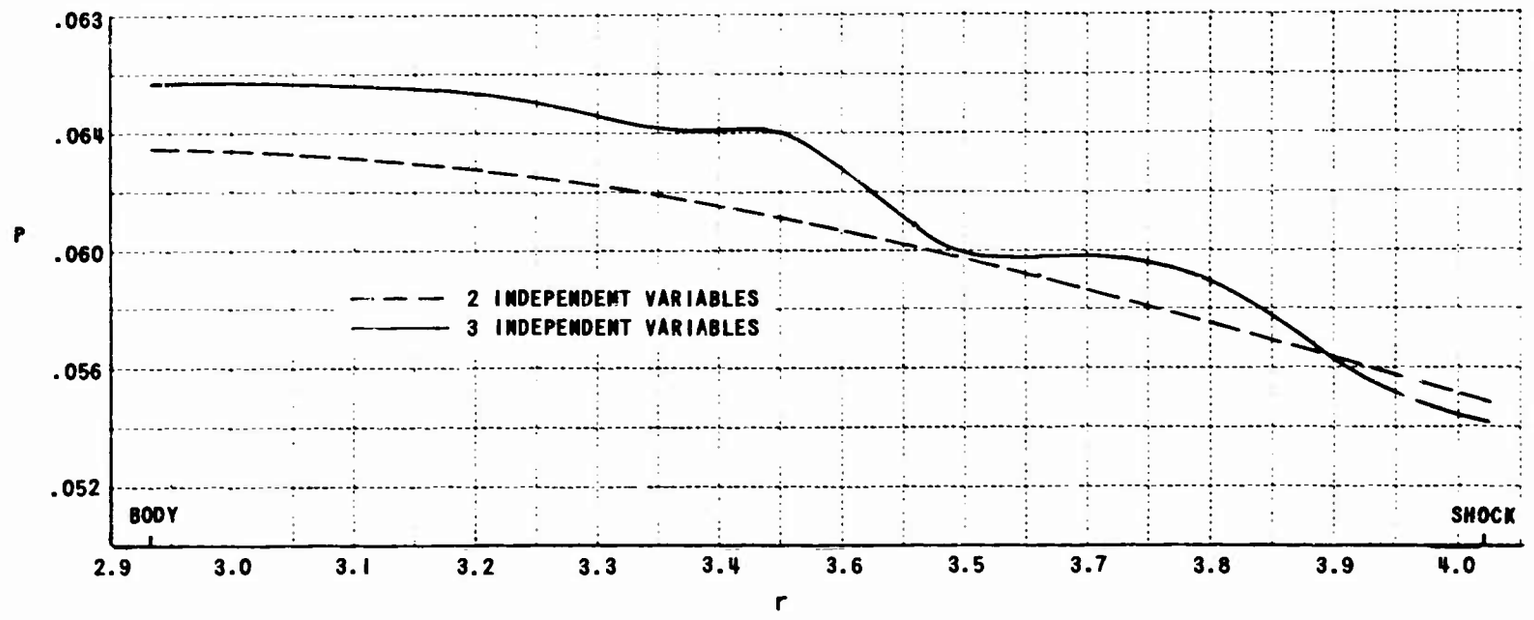


Figure 27 PRESSURE DISTRIBUTION THROUGH SHOCK LAYER BLUNT 15° CONE, α = 0°, X = 7.078

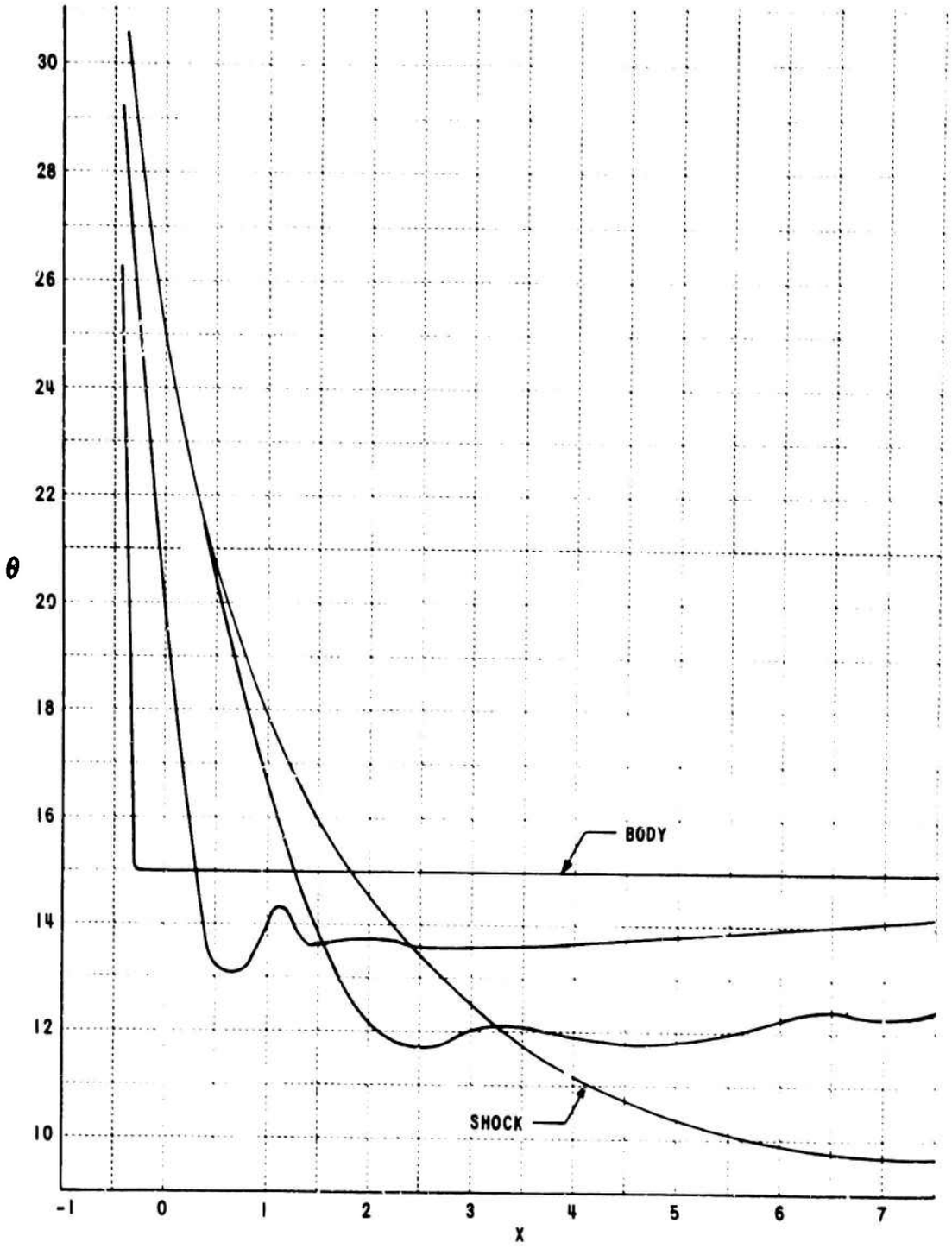


Figure 28 FLOW ANGLES ALONG SELECTED STREAMLINES FOR BLUNT 15° CONE, $\alpha = 0^{\circ}$

expected and there was no noticeable effect on the flow properties along streamlines. The computation proceeded for 0.5 nose radii, during which time, another 3 rings of data points had been added to the field. The overall computation time had thus been decreased by a factor of approximately 4 as compared to the previous rate. The results of the calculation are shown in Figures 29 through 36. Figure 29 shows a side view of the body and the shape of the shock wave on the most windward and most leeward sides of the cone. Also shown are isobaric curves on the body surface. Note the region of transition from axially symmetric flow to fully three-dimensional flow on the body surface. The pressures on the windward and leeward sides of the cone differ by a factor of nearly five at the last axial location calculated.

The three-dimensionality of the flow along the body surface can also be seen in Figure 30, where the azimuthal variation in pressure around the body for several different axial stations is presented. At an axial location $\chi = -0.2$, the pressure is still constant over one half of the body surface. By the time station $\chi = 0.0$ is reached, the flow is completely asymmetric, with the pressure on the leeward surface having fallen by a factor of two. The pressure then continues to decrease around the cone until section $\chi = 1.0$ is reached. Between axial stations $\chi = 1.0$ and $\chi = 2.0$, the pressure near the windward side of the cone has increased while the pressure over the remaining portion of the body surface continues to decrease. When the computation was stopped, the region of increasing pressure encompassed about one-third of the cone surface, and was progressing toward the leeward side as expected. The cone pressure at which the pressures on the windward and leeward sides should stabilize are also shown and appear to be the correct magnitude.

The axial pressure distribution on the most windward generatrix of the cone is shown in Figure 31. The typical overexpansion is evident with the minimum occurring in this case at approximately one nose radius from the center of the sphere. As shown, the pressure is approaching the equivalent cone pressure and should be constant after approximately 5 radii.

The distribution of pressure across the shock layer is shown in Figures 32 and 35 in planes representing four different axial stations. Note

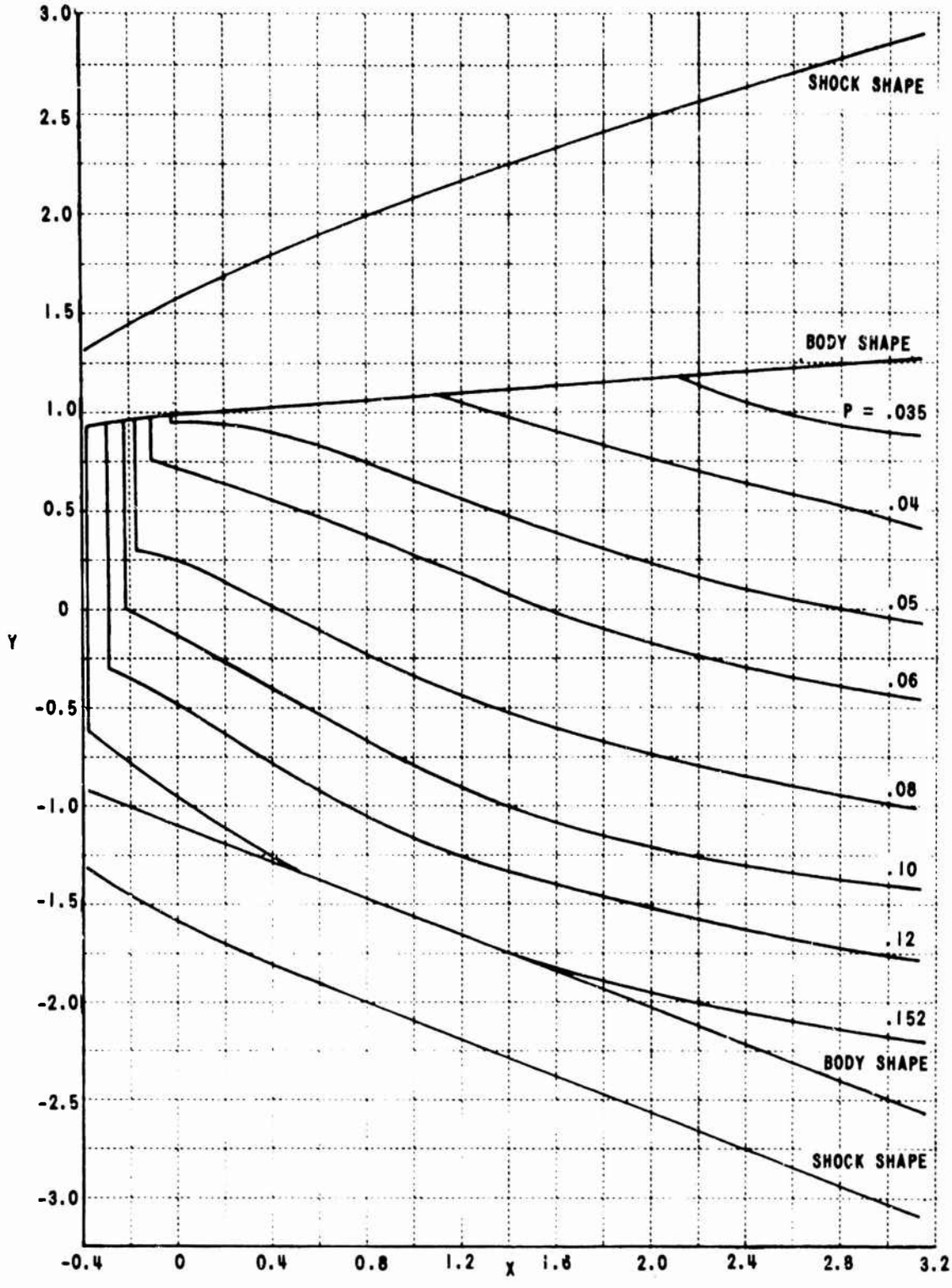


Figure 29 PRESSURE DISTRIBUTION ON BLUNT 15° CONE, α = 10°

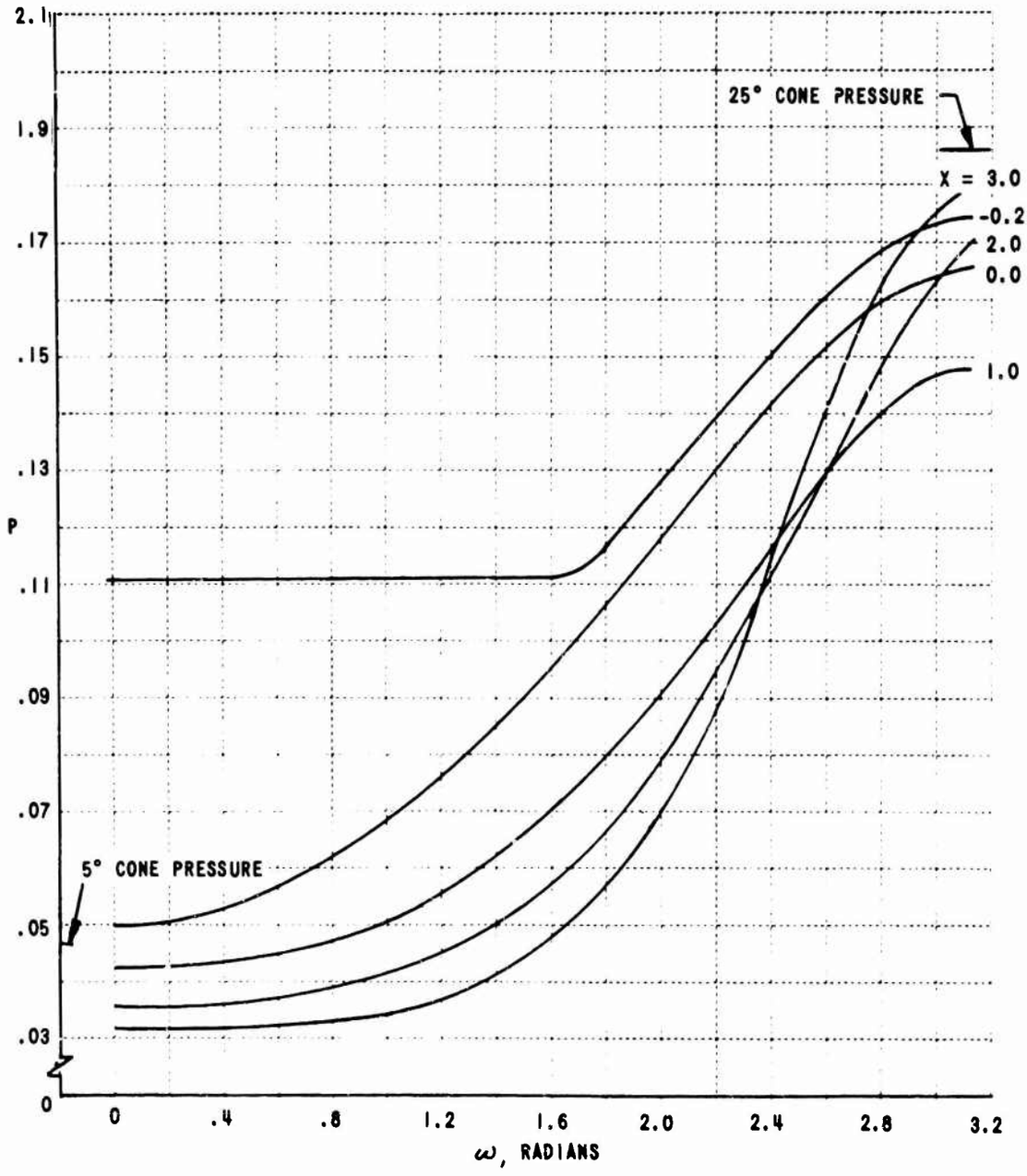


Figure 30 MERIDIONAL PRESSURE DISTRIBUTION FOR BLUNT 15° CONE FOR VARIOUS CROSS SECTIONS, α = 10°

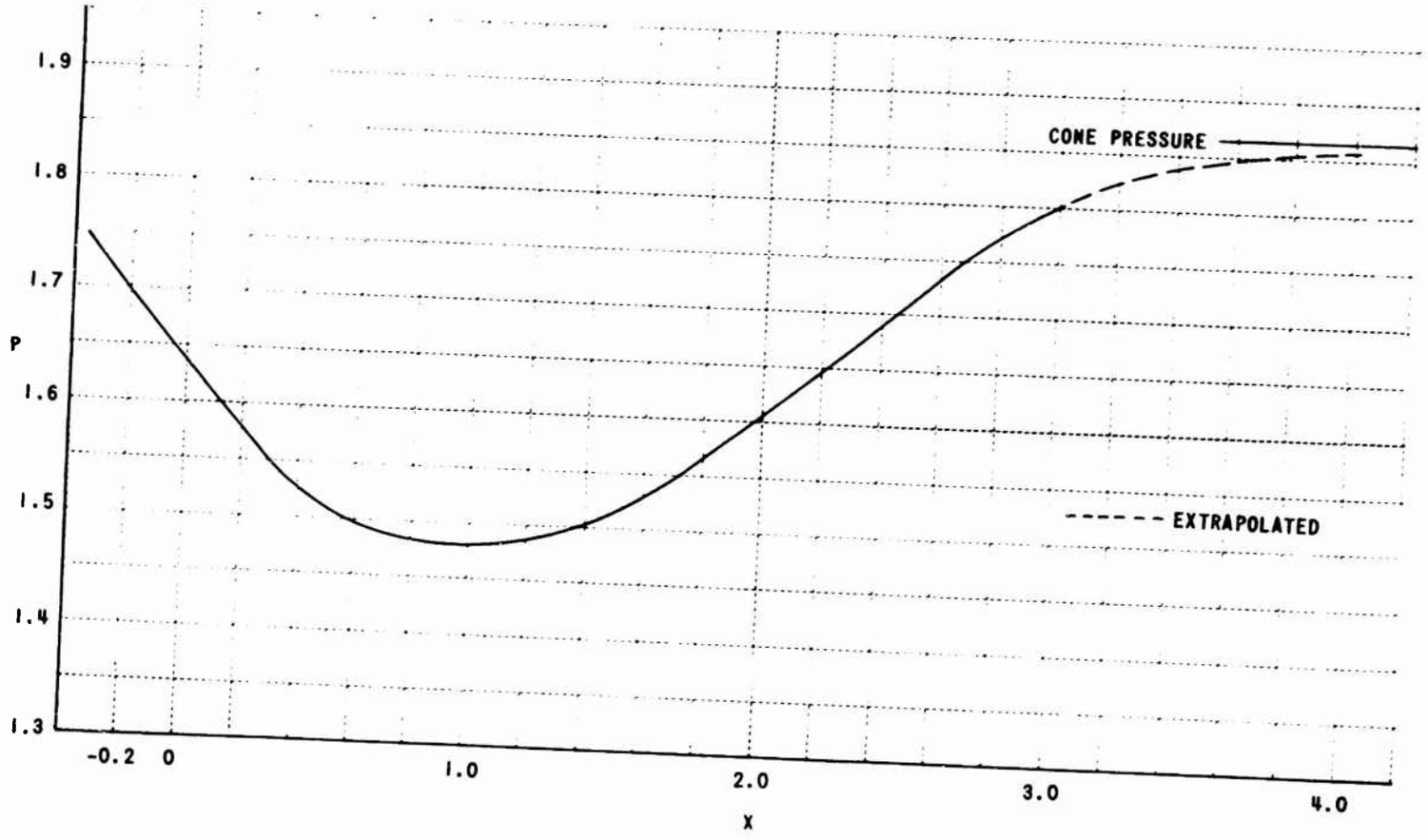


Figure 31 PRESSURE DISTRIBUTION ALONG MOST WINDWARD GENERATRIX - BLUNT 15° CONE, $\alpha = 10°$

that the body shape in each successive plane does not remain circular. Again, this is because the χ-axis is parallel to the free stream velocity vector. The results presented display basically the same pattern as those which occur in cross-sections normal to the body axis.[56] After the non-axisymmetry of the flow has been established, the pressure is nearly constant along rays approximately normal to the body through most of the shock layer. The maximum pressure variation through the shock layer is approximately half the total azimuthal variation on the body at $\chi = 3.124$ with the largest gradients existing near the shock. Note also that the shock wave remains nearly circular with respect to the wind oriented axis, for the region of the flow field calculated. The scale used in Figure 36 is different from that used in the three previous figures. Hence, although the body diameter and shock layer thickness appear to be smaller than those at station $\chi = 2.011$ they are actually much larger.

The pressure distribution obtained along the body surface was smooth, although small oscillations similar to those discussed above for 0° angle-of-attack did appear along rays normal to the body. The amplitudes of the oscillations were in this case much smaller than before, and disappeared when the expansion wave reached the shock. The reason the amplitudes were smaller can probably be attributed to the fact that smaller axial step sizes were taken in the region of the shoulder between the nose sphere and the cone. Hence, the speculation that the oscillations can be completely eliminated if sufficiently small step sizes are taken in this region appears to be justified. The pressure distributions shown in Figures 32 through 35 represent the mean values of the oscillations. A typical oscillation around the mean is shown in Figure 34.

The intersections of various streamlines with the plane $\chi = 3.124$ are shown in Figure 36. The dashed lines represent the locations of the original rays from the body to the shock, as they would appear if the body and flow field were axisymmetric with respect to the origin. For comparison, the shape of the body cross-section as it appeared in the initial plane is also presented. Each curve represents the locus of streamlines which were initially located on the same ray. The streamlines near the body are "spinning"

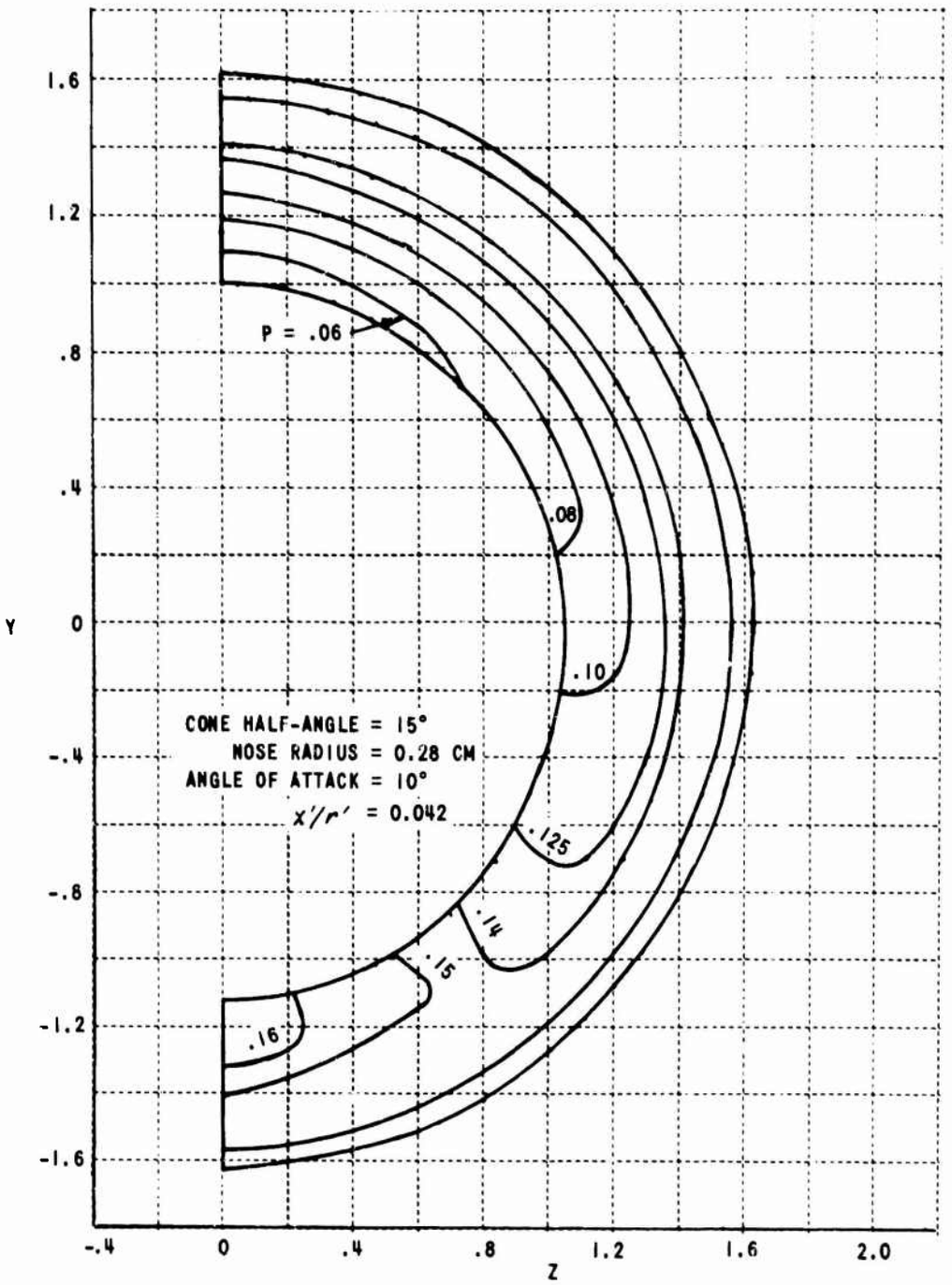


Figure 32 PRESSURE DISTRIBUTION ACROSS SHOCK LAYER FOR SPHERICALLY BLUNTED CONE, $p = p'/\rho_\infty' U_\infty'^2$

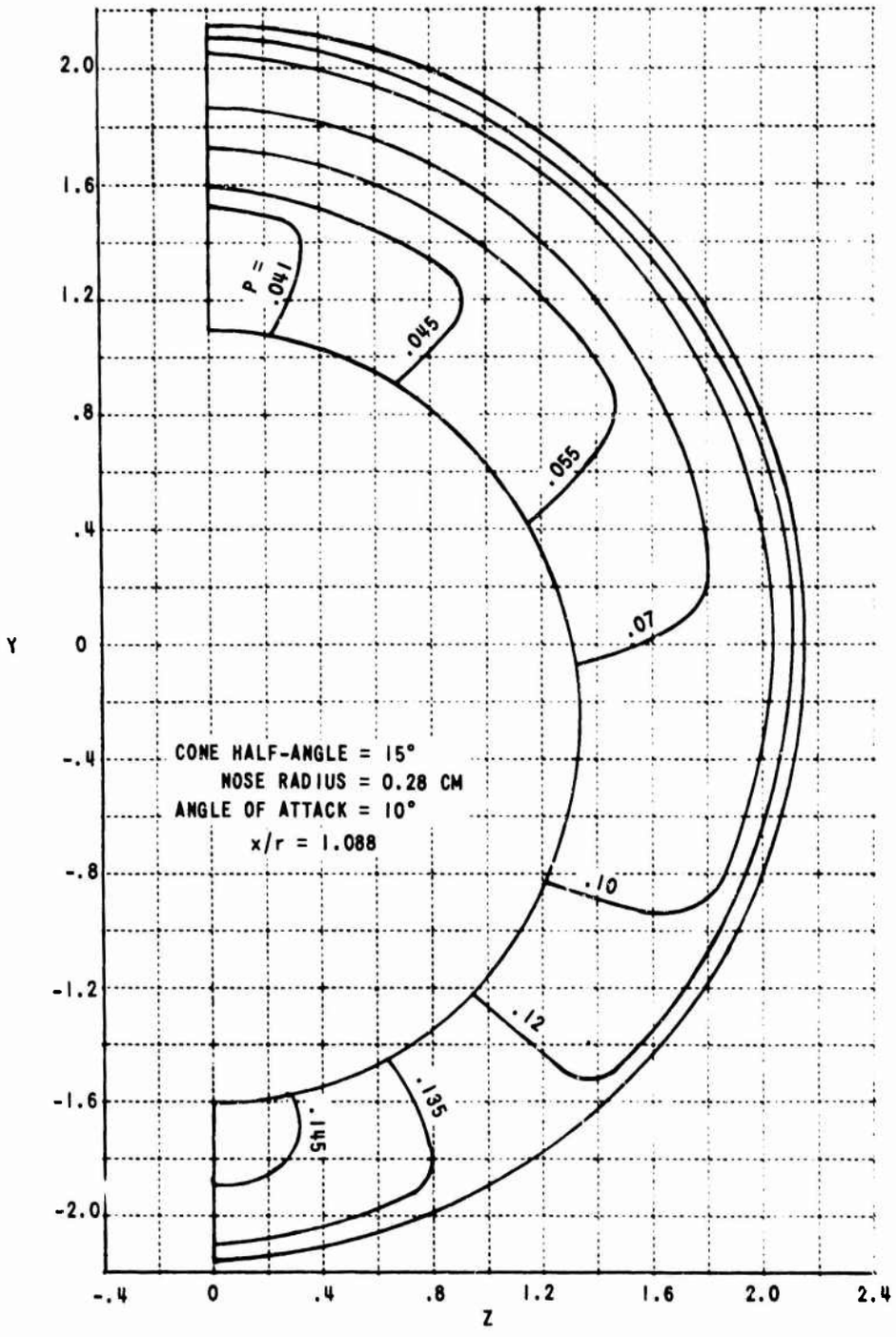


Figure 33 PRESSURE DISTRIBUTION ACROSS SHOCK LAYER FOR SPHERICALLY BLUNTED CONE, $p = p'/\rho_\infty' U_\infty'^2$

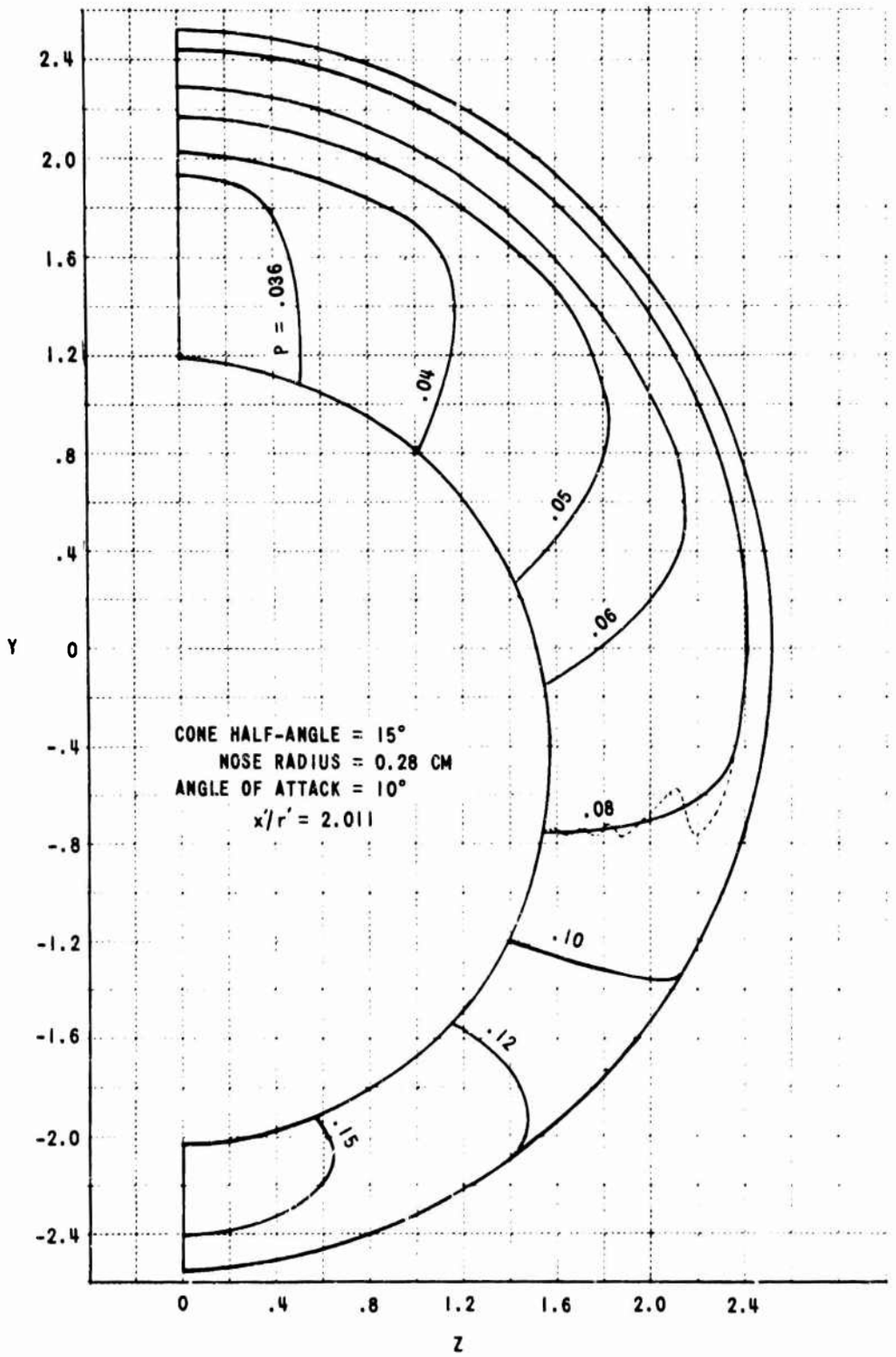


Figure 34 PRESSURE DISTRIBUTION ACROSS SHOCK LAYER FOR SPHERICALLY BLUNTED CONE, $p = p'/\rho_\infty' U_\infty'^2$

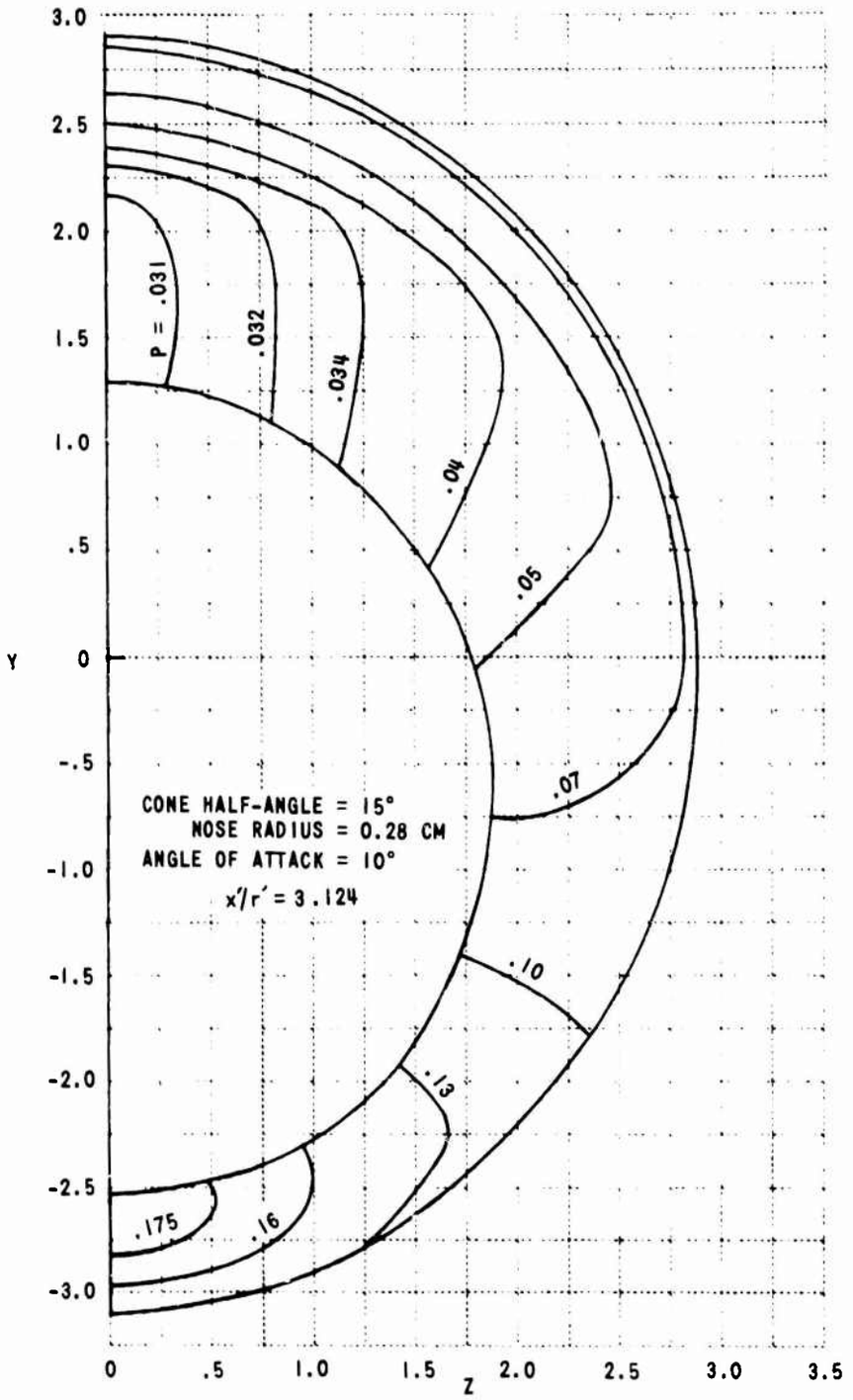


Figure 35 PRESSURE DISTRIBUTION ACROSS SHOCK LAYER FOR SPHERICALLY BLUNTED CONE, $p = p'/\rho_\infty' U_\infty'^2$

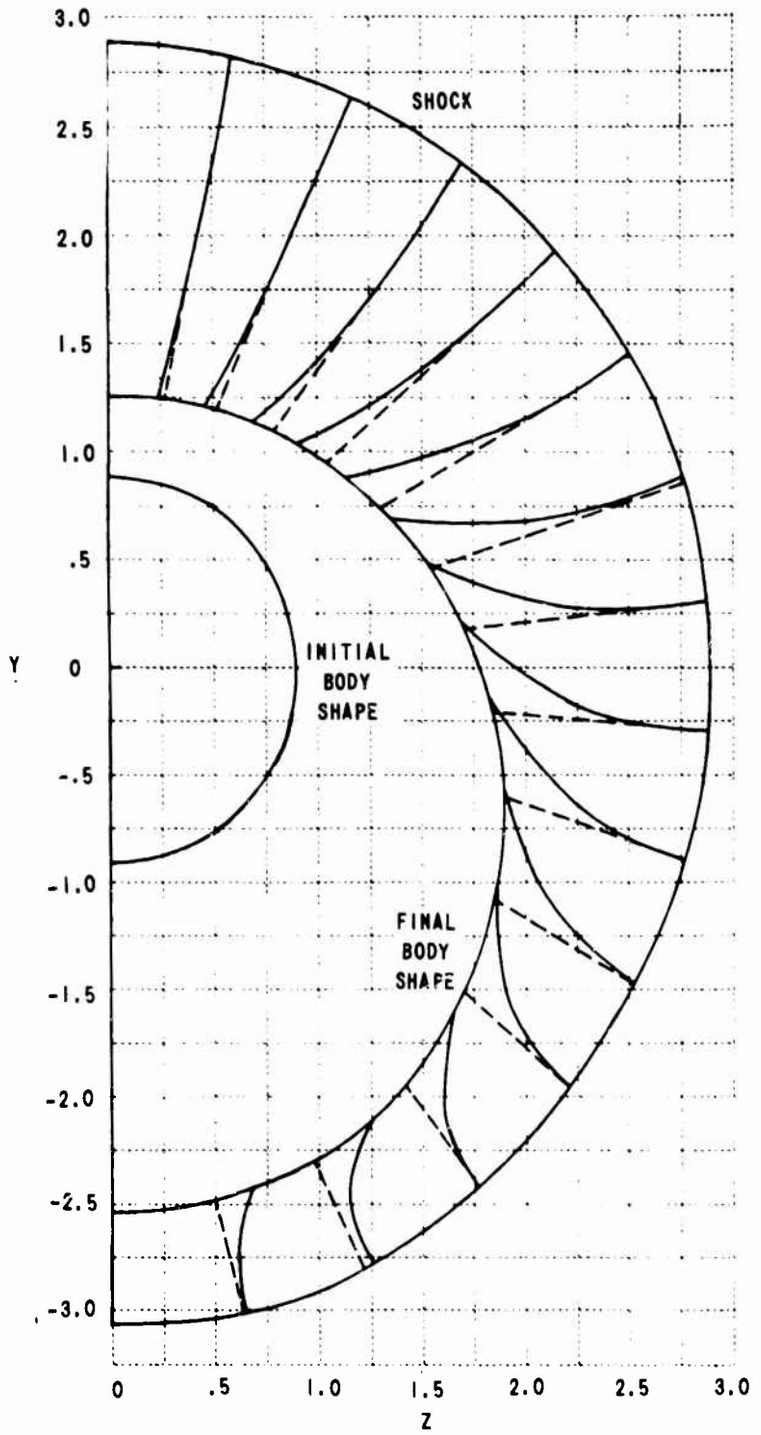


Figure 36 INTEGRATION RAYS IN FINAL PLANE FOR BLUNT 15° CONE, $\alpha = 10°$

toward the leeward side, with the maximum displacements away from the initial ray having occurred on the body in the region where the pressure gradients are greatest, as would be expected. Near the shock however, the displacements are slight, indicating again that the flow in this region has remained nearly axisymmetric. The fact that the streamlines flow toward the leeward side indicates that it will be necessary to include a subroutine in the program which adds new streamlines through the shock layer on the windward side as they are needed for sufficient resolution of the flow field. One convenient way to allow for this, would be to choose more initial rays on the windward side of the body, and use unequal meridional spacings for the rays in the initial plane. The program is capable of handling this situation. Also, it will be necessary to drop streamlines as they crowd together on the leeward side. Probably the best way to accomplish all of these objectives would be to realign the mesh along body normals by interpolation at selected axial stations. Although this procedure involves shifting from one set of streamlines to another midway through the calculation scheme, it can be accomplished with little extra programming effort, since the interpolation routine is already available in the program. The percentage increase in the overall computation time should be small, since the mesh points need be realigned in this manner perhaps only three or four times throughout a particular flow field calculation.

4.3.3 Elliptical Afterbody

The third example consists of the flow field around a spherically blunted elliptical afterbody at zero angle-of-attack. The locus of the semi-major axes of the ellipses in cross-sections normal to the body axis is a straight line inclined at 15° to the body axis, while the locus of the semi-minor axes is a straight line inclined at 5° to the axis. Thus, cross-sections of the body are elliptical with eccentricities varying as a function of the axial coordinate x. The equation of the body is written in three separate sections. First, the spherical nose;

$$B(x, y, z) = x^2 + y^2 + z^2 - (1.0)^2 = 0 \qquad (128)$$

Second, the transition region through which the locus of semi-major axes is a straight line, while the locus of semi-minor axes is a circle;

$$B(x,y,z) = \frac{y^2}{a(x)^2} + \frac{z^2}{b(x)^2} - 1 = 0 \qquad (129)$$

where

$$a(x) = (2x - x^2)^{1/2} \quad .74118 \leq x \leq .91284 \qquad (130)$$

$$b(x) = .7673 + .268x \qquad (131)$$

and finally, the afterbody region where the loci of the semi-major and minor axes are both straight lines and

$$B(x,y,z) = \frac{y^2}{a(x)^2} - \frac{z^2}{b(x)^2} - 1 = 0 \qquad (132)$$

where,

$$a(x) = .91284 + .08749x \qquad (133)$$

$$b(x) = .268x + .7673 \qquad (134)$$
$$x \geq .91284$$

The results of the computation are given in Figures 37 through 40. In this case, a total of 9 meridional rays containing 9 points each was used in the initial plane. The ray spacings were constant and equal to 11.25°. It required nearly 90 minutes on an IBM 7094 to calculate 3 nose radii using an average step of 0.03 radii. Seven rings of data points were added at the shock throughout the entire computation, while no point dropping was employed. Again, since rather large step sizes (0.08) were used in the transition region near the nose, oscillations similar to those discussed above did develop in the field. The curves shown represent the mean values of the oscillations. The amplitudes of these oscillations were again decreasing as the computation proceeded downstream. However, since the initial expansion wave had not yet reached the shock, the oscillations were still apparent. A typical oscillation around an isobar in the final data plane calculated is shown in Figure 40. In this case, the expansion wave is just over one-half the distance through the shock layer.

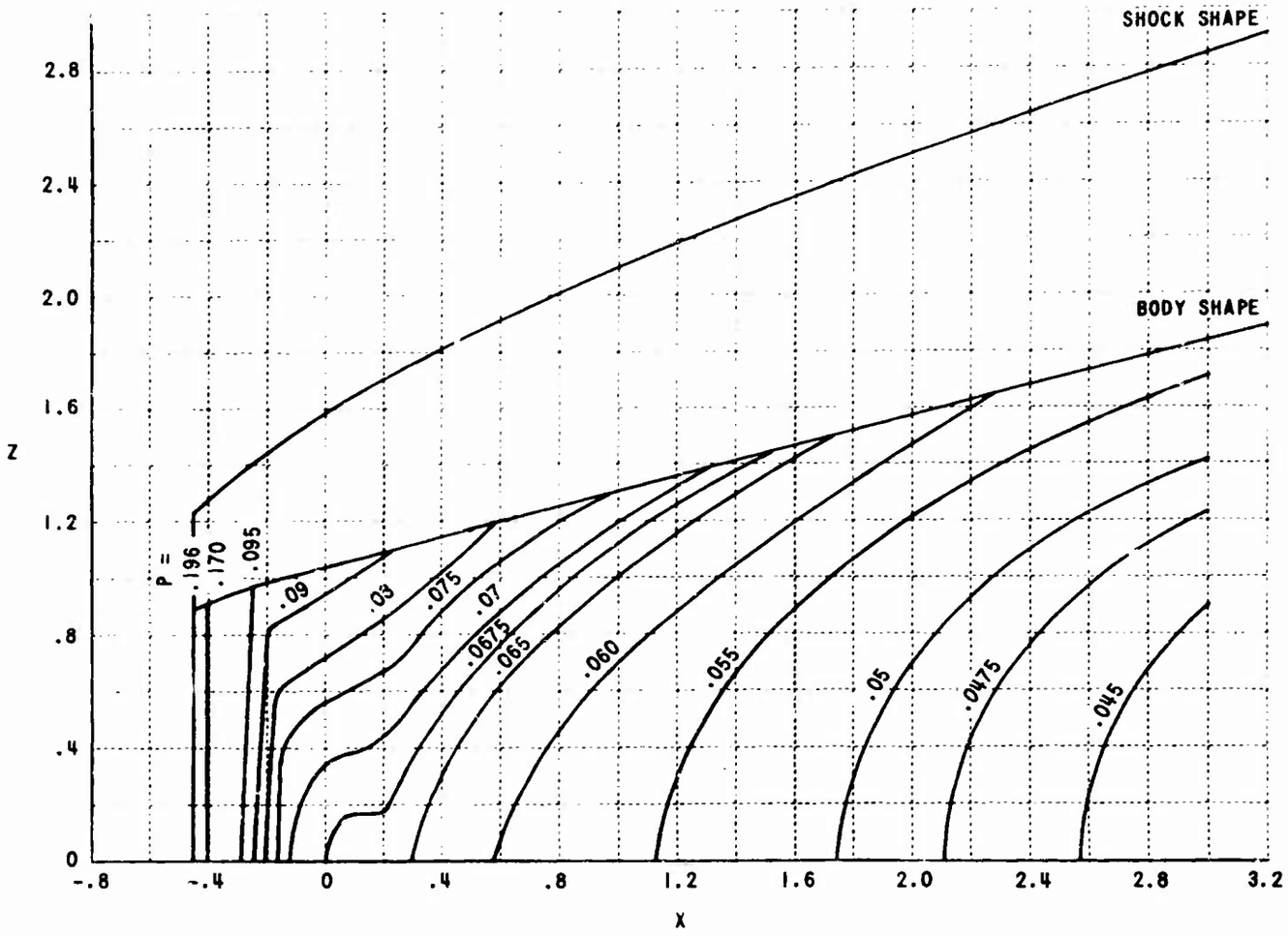


Figure 37 PRESSURE DISTRIBUTION ON BLUNT ELLIPTICAL AFTERBODY, $\alpha = 0°$

The pressure distribution along the body is shown in Figure 37 in the form of isobaric curves. The locus of the semi-minor axes coincides with the χ-axis, while the locus of the semi-major axes is shown in the figure. The asymmetry of the distribution is apparent, with a pressure decrease of 30% occurring around the body in the last plane calculated ($\chi = 3.0$). The "humps" which occur along the isobars near the nose are again caused by the initial overexpansion in the transition region. They could be eliminated by taking smaller step sizes in this region.

The pressure distributions through the shock layer are presented in Figures 38, 39 and 40 for cross-sections at $\chi = 1.07$, 2.01 and 3.02, respectively. The shock wave and body shapes are also shown in each plane. Note that the shock is still axisymmetric in the last plane calculated. Also, the bluntness factor of the body ellipse in the last plane calculated is only 1.2 compared with the asymptotic value of 2.2 for this particular body. Thus, the problem is not nearly as three-dimensional as was intially anticipated. Computer time limitations prevented carrying the solution out farther in the axial direction.

In the last plane, nearly all of the shock layer has an asymmetric pressure distribution, whereas, at $\chi = 1.07$, the distribution is still axi - symmetric through almost one-half of the layer. Along the low pressure edge of the body, the shock layer thickness is 1.3 as compared with the value of 1.0 along the high pressure edge in this final plane. The shock layer thickness is thus over three times as large as the initial value of 0.37. Generally, the pressure distributions correspond to what would be expected for the elliptical afterbody, and except for the oscillations which did arise through the field, the results of the run seem to again verify the fact that the present version of the program is working correctly.

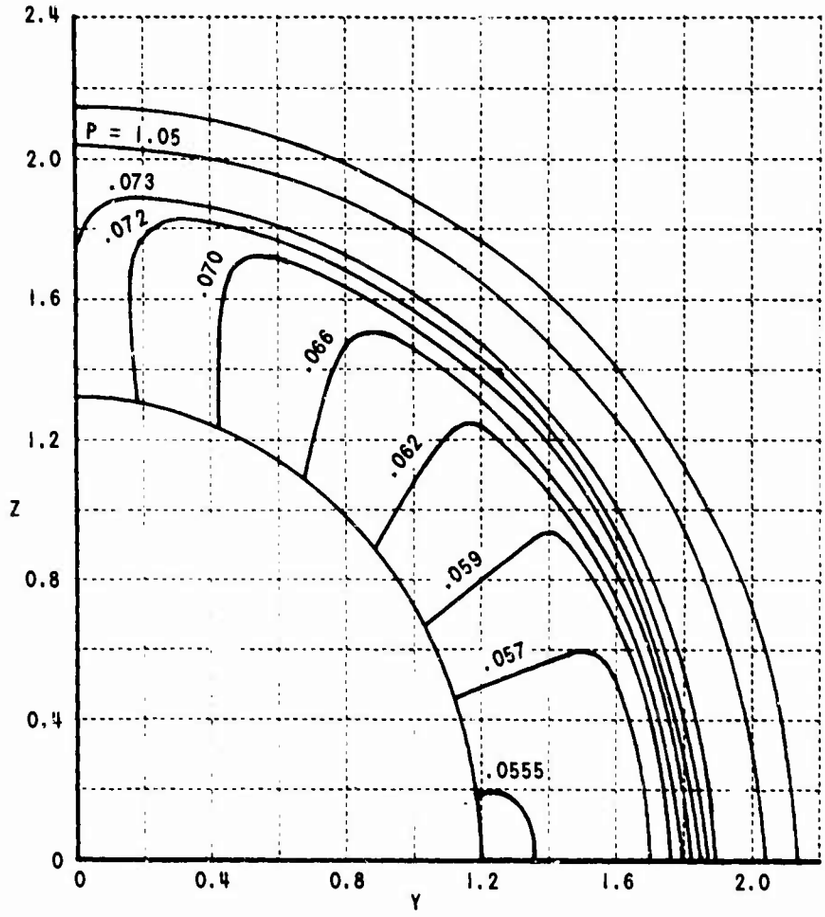


Figure 38 PRESSURE DISTRIBUTION ACROSS SHOCK LAYER FOR SPHERICALLY BLUNTED ELLIPTICAL AFTERBODY, $\alpha = 0°$, X = 1.07

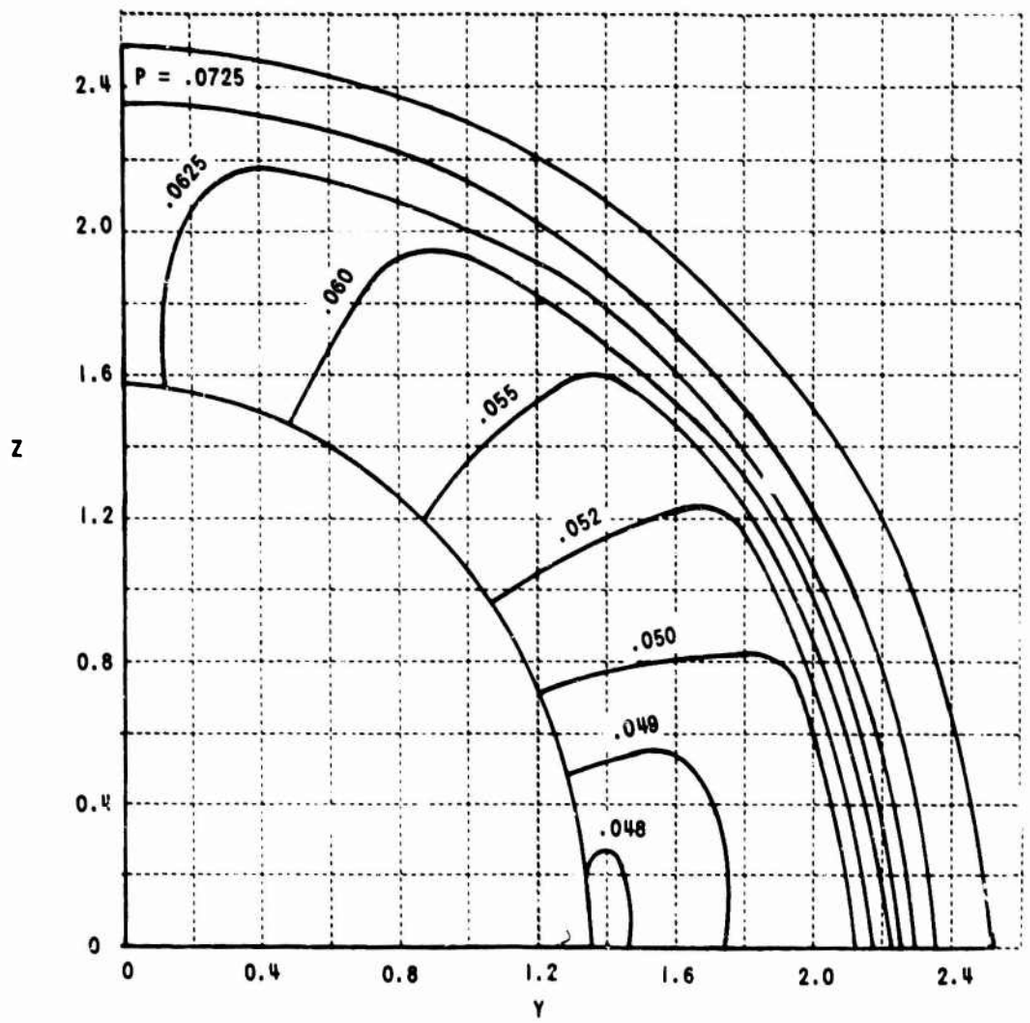


Figure 39 PRESSURE DISTRIBUTION ACROSS SHOCK LAYER FOR SPHERICALLY BLUNTED ELLIPTICAL AFTERBODY, α = 0°, X = 2.014

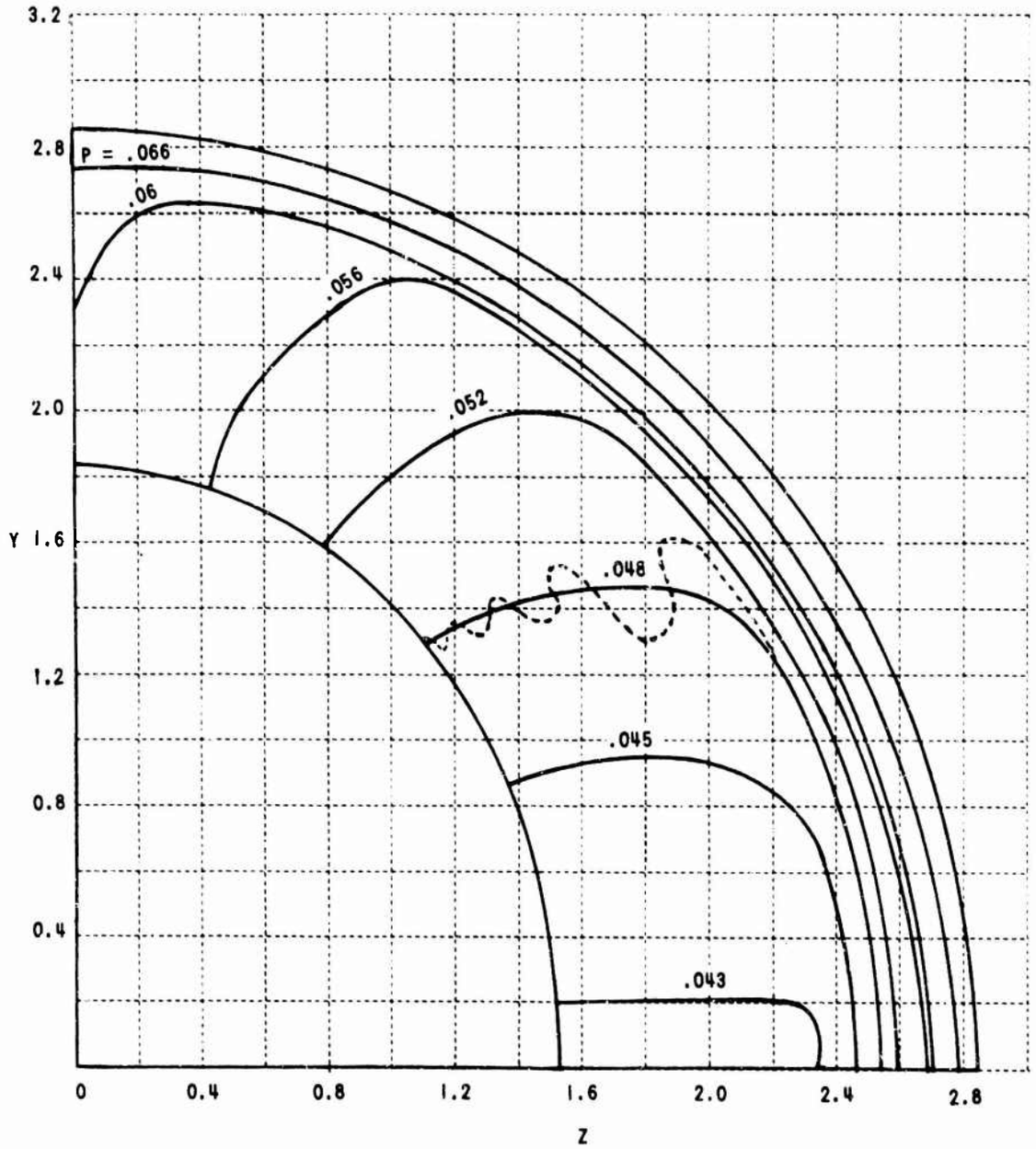


Figure 40 PRESSURE DISTRIBUTION ACROSS SHOCK LAYER FOR SPHERICALLY BLUNTED ELLIPTICAL AFTERBODY, α = 0°, X = 3.018

Section 5

CONCLUSIONS

Many three-dimensional steady flow fields can be calculated utilizing the numerical procedures presented in this report; however, the computer times are quite long. Extension of the program developed here to include physico-chemical models more complex than the perfect gas can also be readily accomplished. As a matter of fact, the scheme developed is especially applicable to solving chemically frozen and finite-rate reacting flows. However, the economic practicality of such calculations is considered to be marginal with presently available equipment.

On the next generation of computers these problems should be feasible. The three-dimensional steady flow program developed here involve large amounts of input and output data. Although the abundance of output data can be controlled by printing out only a few selected planes of data during a computation, the tasks of handling and presenting even these data are still tedious. The development of internal computer methods for quickly handling and displaying the output data graphically or in some other convenient form, is thus necessary.

Several basic conclusions can be stated regarding the general numerical scheme. First, the Courant-Friedrichs-Lewy (C-F-L) stability criterion proved to be a sufficient condition to insure stability of the integration procedure. It should be noted, however, that with the fitting procedures utilized here, strict adherence to the C-F-L condition is probably not always required. Hence, it appears possible to enlarge the domain of dependence of the difference equations slightly by extrapolation using the polynomial surface fits without affecting the accuracy of the solution. The total computing time per problem would in turn be decreased. Second, for accuracy and stability, higher order fitting procedures than linear are necessary. The quadratic fits used here are expected to be sufficiently accurate for most flow problems. Finally, in order to reduce truncation errors to second order in the step size, it is necessary to add extra bicharacteristics in addition to those that would be required for a simplical mesh. The results obtained from the program utilizing the extra bicharacteristics compared closely with those obtained from a two independent variable method of characteristics solution.

The greatest advantage to be derived from the procedure developed here lies in the fact that for the most part (except when mesh "squaring" is used) the integrations proceed along the same streamlines. This greatly simplifies inhomogeneous-frozen and nonequilibrium flow field calculations by significantly reducing the number of interpolations. Having the output data available along streamlines is also useful for visualizing the flow field associated with a particular body, and for applying the pressure distribution obtained from the program to calculate three-dimensional boundary layers.

For the Cauchy problem considered, a large amount of initial data is required to calculate a given flow field. For a general three-dimensional body, this data must be supplied at points on an initial plane (or surface) located entirely in the supersonic portion of the flow field. Thus, what is in general required is some method for obtaining asymmetric inputs on the initial plane. Unfortunately, such a method is not yet available, and, for the problems considered, it is necessary to limit the geometries to spherically capped afterbodies, for which axisymmetric inputs can be utilized. When methods for calculating non-axisymmetric blunt body flows become available, they can be used to supply inputs to the steady flow method of characteristics scheme utilized here. One possible way to obtain asymmetric inputs would be to generalize the present program to include unsteady flow. Starting from one known steady blunt body solution, another steady flow solution could be obtained asymptotically in time, after the nose cap has undergone certain prescribed motions (or shape changes). Thus, using a combination of three-dimensional steady and unsteady program, one could analyze rather general body shapes and motions.

Section 6

REFERENCES

1. Telenin, G. F. and Tinyakov, G. P., "Method of Computing Three Dimensional Flows Past Solid Bodies after Shockwave Passage," Dokl. A.N. SSSR, Gridromekhanika, Tom 154, No. 5, February 1964, p. 1056-1058. (NASA Technical Translation, P.8682 by A. L. Brichant, April 1964)

2. von Neumann, J. and Richtmyer, R. D., "A Method for the Numerical Calculation of Hydrodynamic Shocks," Journal of Applied Physics, Vol. 21, 1950, pp. 159-193.

3. Burstein, Samuel Z., "Numerical Calculations of Multidimensional Shocked Flows," A.I.A.A. Preprint No. 64-2, January 1964.

4. Lax, Peter D., "Weak Solutions of Nonlinear Hyperbolic Equations and their Numerical Computation," Commun. on Pure and Applied Math., Vol. VII, 1954, pp. 159-193.

5. Lax, P. D. and Wendroff, B., "Systems of Conservation Laws," Commun. Pure and Applied Math., Vol. 13, 1960, pp. 217-237.

6. Crocco, L., "Solving Numerically the Navier Stokes Equations," General Electric Co. Report 63SD891, May 1960.

7. Forsythe, G. E. and Wasow, W. R., Finite-Difference Methods for Partial Differential Equations, Wiley, New York, 1960.

8. Fox, L., Numerical Solution of Ordinary and Partial Differential Equations, Addison-Wesley Publishing Co., Inc., 1962.

9. Sauerwein, H., "The Calculation of Two- and Three-Dimensional Inviscid Flows by the Method of Characteristics," M.I.T. Fluid Dynamics Research Laboratory Report, June 1964.

10. Coburn, N. and Dolph, C. L., "The Method of Characteristics in the Three-Dimensional Stationary Supersonic Flow of a Compressible Gas," Proceedings of Symposia in Applied Mathematics, Vol. 1, 1949, pp. 55-56.

11. Titt, E. W., "An Initial Value Problem for all Hyperbolic Partial Differential Equations of Second Order with Three Independent Variables," Ann. Math (2), Vol. 40, 1939, pp. 862-891.

12. Ferri, Antonio, "The Method of Characteristics," Section G in General Theory of High Speed Aerodynamics, Vol. VI of High Speed Aerodynamics and Jet Propulsion, Princeton University Press, 1954.

13. Meyer, R. E., "The Method of Characteristics," in Modern Developments in Fluid Dynamics High Speed Flow, ed. by L. Hawarth, Oxford Univ. Press, London, 1953, pp. 71-104.

14. Thornhill, C. K., "The Numerical Method of Characteristics in Three Independent Variables," Aeronautical Research Council R. & M. No. 2615, September 1948.

15. Clippinger, R. F. and Giese, J. H., "Characteristic Conditions for Three Dimensional Flows with Vorticity," Ballistic Research Laboratories Memorandum Report 615, Aberdeen Proving Ground, Maryland, August 1952.

16. Holt, M., "The Method of Characteristics for Steady Supersonic Rotational Flow in Three Dimensions," JFM, Vol. I, Part 4, October 1956, p. 409.

17. Fowell, L. R., "Flow Field Analysis for Lifting Re-entry Configurations by the Method of Characteristics," I.A.S. Paper No. 61-208-1902, June 1961.

18. Ferrari, C., "Interference Between Wing and Body at Supersonic Speeds - Analysis by the Method of Characteristics," JAS, Vol. 16, 1949, pp. 411-434.

19. Moeckel, W. E., "Use of Characteristic Surfaces for Unsymmetrical Supersonic Flow Problems," NACA TN. 1849, March 1949.

20. Bruhn, G. and Haack, W., "Ein Charakteristikenverfahren für Dreidimensionale Instationare Gasstromungen," Z. angew. Math. Phys., Vol. 9, 1958, pp. 173-190. (trans. by C. Strom)

21. Butler, D. S., "The Numerical Solution of Hyperbolic Systems of Partial Differential Equations in Three Independent Variables," Proceedings of the Royal Society, Series A, Vol. 255, 1960, pp. 232-252.

22. Tsung, Cheng-Chih, "Study of Three-Dimensional Supersonic Flow Problems by a Numerical Method Based on the Method of Characteristics," Ph.D. Thesis in Mech. Eng., University of Illinois, 1960.

23. Moretti, G., Sanlorenzo, E. A., Magnus, E.E., and Weilerstein, G., "Supersonic Flow about General Three-Dimensional Blunt Bodies," Vol. III, Flow Field Analysis of Reentry Configurations by General Three-Dimensional Method of Characteristics, ASD-TR-61-727, 1962.

24. Kackova, O. N. and Cuskin, P. I., "A Scheme for the Numerical Method of Characteristics," Soviet Math. Doklady, Vol. 5, No. 1, January-February 1964.

25. Chu, Boa-Teh, "Wave Propagation and the Method of Characteristics in Reacting Gas Mixtures with Applications to Hypersonic Flow," AD-118350, WADC TN-57-213, Division of Engineering, Brown University, Providence, Rhode Island, 1957.

26. Resler, E. L., "Characteristics and Sound Speed in Nonisentropic Gas Flows with Nonequilibrium Thermodynamic States," J. A. S., Vol. 24, No. 11, November, 1957, pp. 785-790.

27. Wood, W. W. and Kirkwood, J. G., "The Hydrodynamics of a Reacting and Relaxing Fluid," J. Appl. Phys., Vol. 28, No. 4, April 1957, pp. 395-399.

28. Sedney, R., South J. C. and Gerber, N., "Characteristic Calculation of Nonequilibrium Flows," Aberdeen Proving Ground, BRL Report 1173, April 1962.

29. Sedney, R. and Gerber, N., "Nonequilibrium Flow Over a Cone," IAS Preprint 63-40, 1963.

30. Capiaux, R. and Washington, M. "Nonequilibrium Flow Past a Wedge," IAS Preprint 63-71, 1963.

31. Eastman, D. W., "Two Dimensional or Axially Symmetric Real Gas Flows by the Method of Characteristics, Part IV: Nonequilibrium Flow," AD-438263, Boeing Airplane Co. Report No. D2-20874, September 1962.

32. Zupnik, T. F., Nilson, E. N., Landis, F., Keilbach, J. R. and Ables, D., "Application of the Method of Characteristics Including Reaction Kinetics to Nozzle Flow," A. I. A. A. Preprint No. 64-97, January 1964.

33. Widawsky, A., "Computer Program for the Determination of Chemically Reacting Flow Fields by the Method of Characteristics," AD-437698, Technical Documentary Report No. RTD-TDR-63-4286, October 1963.

34. Wood, A. D., Springfield, J. F. and Pallone, A. J., "Determination of the Effects of Chemical and Vibrational Relaxation on an Inviscid Hypersonic Flow Field, AVCD RAD-TM-63-88, January 1964.

35. Curtis, J. T., Burke, A. F., Wallace, J. E. and Butler, F. E., "Numerical Studies of Hypersonic Flow Fields Including the Effects of Finite-Rate Chemical Reactions and Boundary Layer Interaction," CAL Report No. AA-1747-Y-1, April 1964.

36. Courant, R. and Hilbert, D., Methods of Mathematical Physics, Vol. II, Partial Differential Equations, Interscience Publishers, New York and London, 1962.

37. Hadamard, J., Lectures on Cauchy's Problem in Linear Partial Differential Equations, Book I, General Properties of Cauchy's Problem, Oxford University Press, London, 1923.

38. von Mises, Richard, Mathematical Theory of Compressible Fluid Flow, Vol. 3 of Applied Mathematics and Mechanics, Academic Press, New York, 1958.

39. Marrone, P. V. and Garr, L. J., "Inviscid Nonequilibrium Flow Behind Bow and Normal Shock Waves," Part I and Part II, AD-408776, AD-410247, Report No. QM-1626-A-12, May 1963.

40. Courant, R., Friedrichs, K. O. and Lewy, H., "Uber die Partiellen Differenzengleichungen der Mathematischen Physick," Math. Ann., Vol. 100, 1928, pp. 32-74.

41. Sauerwein, H. and Sussman, M., "Numerical Stability of the Three-Dimensional Method of Characteristics," A.I.A.A. Journal, Vol. 2, No. 2, February 1964.

42. Talbot, G. P., "Stability Considerations for the Numerical Method of Characteristics in Three Independent Variables," A.R.D.E. Memorandum (B) 77/60, November 1960.

43. Sauer, R., "Application of Characteristic Theory for Partial Differential Equations to the Solution of Problems Arising in Three-Dimensional Flow," Zeitschrift für Angerwandte Mathematik und Mechanic, Vol. 30, November 1950, pp. 347-356.

44. Ferri, A., "Supersonic Flow Around Circular Cones at Small Angles of Attack," NACA Technical Note 2236, 1950.

45. Ferrari, Carlo, "On the Application of the Method of Characteristics to the study of Irrotational Three-Dimensional Supersonic Flow," CAL Report No. 53, May 1953.

46. R. Sauer, R. Schaetz, Project Report TH Muenchen 1962

47. Treanor, C.E., "A Method for the Numerical Integration of Coupled First Order Differential Equations with Greatly Different Time Constants," Cornell Aeronautical Laboratory Report No. AG-1729-A-4, 1964.

48. Lapidus, Leon, Digital Computation for Chemical Engineers, McGraw-Hill Book Company, Inc., New York, 1962.

Curtis, J. T., "Computer Techniques for Calculating the Flow of a Reacting Gas by the Method of Characteristics," Cornell Aeronautical Laboratory Report No. AA-1577-Y-3, October 1962.

50. Tate, S. E., "An Analysis and Calculation Procedure for Gasdynamic Processes in Thermal and Chemical Equilibrium," AD-246521, Cornell Aeronautical Laboratory Report No. AD-1345-W-4, September 1960.

51. Boyer, D., Personal Communication.

52. Belotserkovskii, O. M., "Flow Past a Circular Cylinder with a Detached Shock Wave," Vychislitelnaia Mathematika, Vol. 3, pp. 149-185, 1958.

53. Bohachevsky, I. O., Personal Communication.

54. Thacher, H. C. Jr., and Milne, W. E., "Multivariate Interpolation," Journal of the Society of Industrial and Applied Mathematics, No. 8, 1960, p. 33.

55. Milne, W. E., Arntzen, W., Reynolds, N., and Wheelock, J., "Mathematics for Digital Computers, Volume I, Multivariate Interpolation," AD-131033, WADC Technical Report No. 57-556, ry 1958. February 1958.

56. Anon., "Pressure Distribution on a Blunted 15° Circular Case," General Applied Science Laboratory Report No. 136, 1960.

57. Burke, A., Personal Communication.

58. Anon., "Equations, Tables, and Charts for Compressible Flow," NACA Report 1135, 1953.

59. Strom, Charles R., The Method or Characteristics for Three Dimensional Steady and Unsteady Reacting Gas Flows, PhD Thesis, University of Illinois, 1965.

60. Curtis, J. and Strom, C., Computations of The Nonequilibrium Flow of a Viscous, Radiating Fluid About A Blunt Axisymmetric Body, Vol. I Equations and Results. AFFDL-TR-67-40. Vol. II Program Details and Operating Procedure.

61. Moretti, G. and ABBETT, M., A Fast Direct, and Accurate Technique For the Blunt Body Problem, GASLTN No. 583, Part I: Analysis, Jan. 31, 1966.

62. Ludford, G., Polachek, H., and Seeger, R. J. "On Unsteady Flow of Compressible Viscous Fluids," Journal of Applied Physics, Vol. 24, April 1953, pp. 490-495.

63. Strang, W.G., "The Expansion of a Gas Cloud Into a Vacuum, Part II, Numerical Solution of the Eulevian Gas Expansion Equations," AD607962, Space Technology Labs., Inc. Report TR-60-0000-00009 (1960)

Appendix A
PROGRAM DETAILS AND OPERATING PROCEDURES

This appendix presents the information necessary for running the program and interputing the output. Paragraph A. 1 contains information on the program limits and its application and tape requirements. Paragraph A. 2 gives detailed description of the input data and its preparation for starting or restarting a given computational problem. Also included here are prepared input format sheets for ease in punching the input data cards. The output from the program is discussed in Paragraph A. 3. Details concerning the data storage are given in Paragraph A. 4. A description of each subroutine is included in Paragraph A. 5. A correspondence table listing the correspondence between the FORTRAN and the report symbols is included in Paragraph A. 6. Finally, Paragraph A. 7 includes logical flow charts describing the program.

A. 1 DESCRIPTION OF DIGITAL COMPUTER PROGRAM

The computer program described here is an extension and modification of that developed in Reference 59. It was originally intended that the final program would be capable of computing supersonic flows around fairly general bodies including exactly the effects of chemical reactions and vibrational relaxations. However, experience with the ideal gas three variable program and axisymmetric and two-dimensional solutions (see Reference 60) indicated that such computations were beyond the capabilities of present computer systems when reasonable running times and machine storage requirements were taken into account. Therefore, in the interest of conserving machine time while at the same time obtaining a reasonably close approximation to the flow field, it was decided that the next logical and least costly extension to the ideal gas program would be to add the frozen-inhomogeneous thermo-chemical model described above. This modification was introduced into the present version of the program. The pressure distribution thus obtained corresponds to the frozen solution.

It may then be utilized to obtain a more complete description of the flow field by using a one dimensional stream tube integration program along each of the streamlines followed in the characteristics procedure. To facilitate the computations for the nonequilibrium solution, the streamline locations and pressures for each step are printed out on binary tape so that they may then be easily used by a suitable stream tube integration program. Actually such a computation has been performed but the program required is not yet completed and is not being reported here. Since the present version of the program is considerably different than the original ideal gas version, the solutions described above and in Reference 59 were recomputed for an ideal gas. In this manner many program errors were uncovered and corrected. The program was also run using the frozen-inhomogeneous option for a 6 inch nose radius spherically capped cone at zero angle of attack. The results indicated that this option is working correctly.

A.1.1 General Discussion

The greatest part of the computations involved in determining the flow properties in the shock layer about a three-dimensional body are contained in the subroutines which calculate body points, field points and shock points in the manner set forth in Section 3. All other subroutines (including the main program) are merely subordinate routines to these three and are used to shift required data around in storage, perform supporting calculations and handle input/output procedures.

The main program controls the overall logic of the scheme by first choosing an initial point located in the field or on the body or shock surfaces, calling the respective principal subroutine and accepting the results for the newly calculated point. It then stores these results as a point on the new data plane and proceeds on to another initial point. When all the points on the initial data plane are thus determined the newly calculated data plane then serves as the next initial plane, and the calculations are continued. New data planes normal to the X-direction are continually calculated until either a predetermined coordinate is reached, or a given maximum number of planes are so determined -- at which time the computations are halted. The relationships among the various subroutines used and the main program are shown in terms of a general calling sequence chart in Figure A-1. A detailed description of each of the subroutines will be given in Paragraph A. 3.

A.1.2 Program Versions

Due to the length of the program, it is necessary to employ either the CHAIN feature of the 7044 operating system or an overlay structure on the 7094 (see Figure A-2). The differences between the two operating systems has necessitated the preparation of two closely similar versions of the program. The version described here is that suitable for the 7094. For the 7044 system the subroutine INPUT is treated as a separate link in the CHAIN.

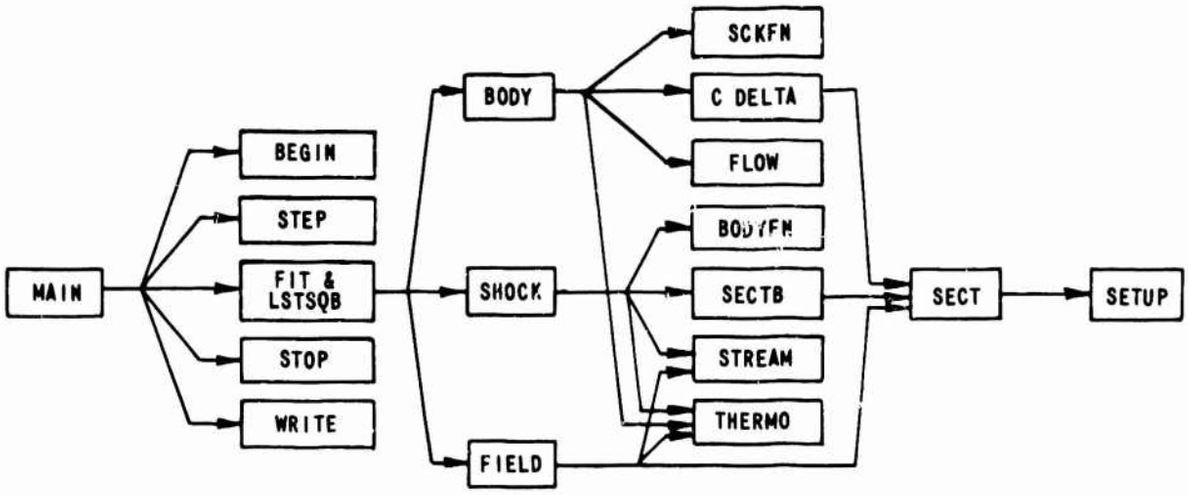
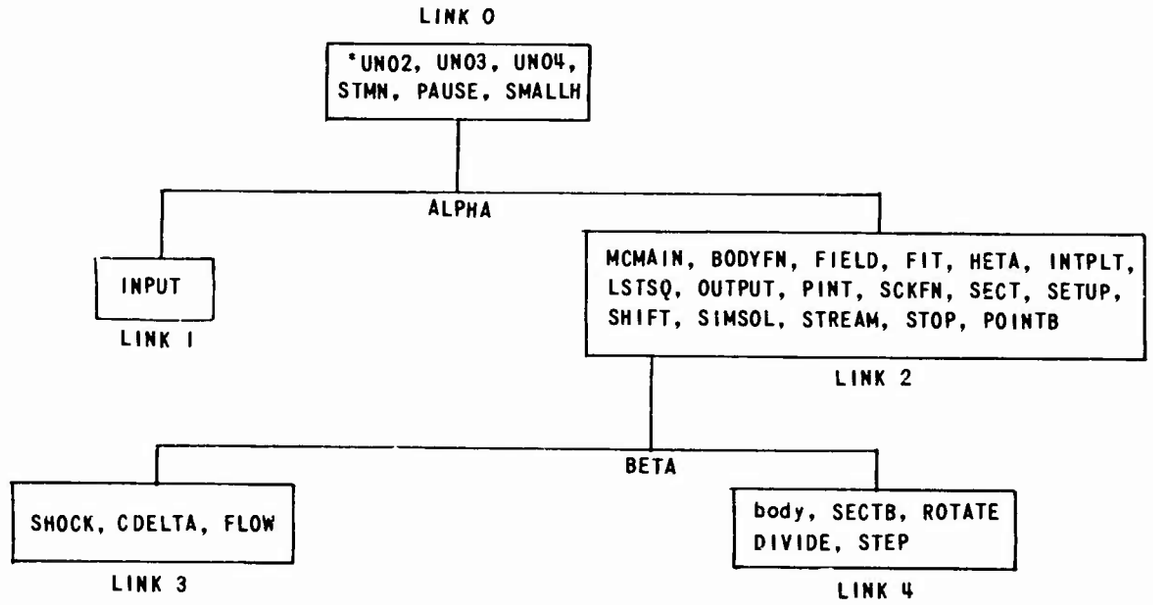


Figure A-1 SUBROUTINE CALLING SEQUENCE



*UNO2, 3,4 SETUP OUTPUT TAPES.

Figure A-2 OVERLAY STRUCTURE FOR IBM 7094

A.1.3 Tape Requirements

(a) TAPE 2 binary tape. Data from an integration plane are stored on tape 2 for every N planes computed where N is an input constant. This tape is used for restarting a computation.

(b) TAPE 3 binary tape. Common data are stored on tape 3. This may be used to provide inputs to a finite rate stream tube integration program.

(c) TAPE 4 binary tape. Streamline data which can be used for reacting stream tube integration scheme is stored on binary tape 4.

(d) TAPE 5 System input tape.

(e) TAPE 6 System output tape.

A.2 DESCRIPTION AND PREPARATION OF INPUT DATA

As was discussed in Paragraph 3.5, a plane (or surface) of inputs is required to any three-dimensional characteristics program. Obtaining data for such an initial plane is sometimes difficult owing to the lack of suitable three-dimensional blunt-body solutions. However, for many configurations having spherical noses, it is possible to assume axisymmetric flow up to some axial station. For such cases, the program described in Reference 60 may be modified to punch out most of the input cards needed for the present program. The necessary changes are detailed in Appendix C.

In this report, the inputs were obtained by combining an inverse technique[39] with an axisymmetric method of characteristics solution[35]. The inverse scheme of Garr and Marrone was used to obtain several data points between the shock and the body on a line normal to the shock and located slightly downstream of the sonic line. The two variable method of characteristics program was then utilized to extend this solution a short distance downstream, until a sufficient number of initial data points could be determined on a line normal to the free stream wind direction (see Figure 19). A plane normal to the wind direction containing this line is chosen to be the initial data plane for the three-dimensional computation. Since the body is initially axisymmetric (e.g., a spherical nose cap) it suffices to obtain the input along one line only. Flow properties at initial data points throughout the initial plane are obtained by rotating the initial data line around the x axis (e.g., the wind axis) in chosen increments in ϕ, the angle between any azimuthal plane and the xy plane. At each point chosen on the initial data line, the radius r, the pressure p, the velocity q, and the two-dimensional flow angle χ are required as inputs to the program. In addition, the two-dimensional shock angle σ, the angle-of-attack α, the gas constant R, and various free stream and nondimensionalizing parameters are needed. All input parameters are dimensioned such that the following form of the equation of state holds

$$p' = \rho' \frac{R'}{MW'} T' \qquad \text{(A-1)}$$

for whatever units are chosen. Energies are made nondimensional with relations of the form

$$h = \frac{h' M N_\infty'}{R' T_\infty'} \tag{A-2}$$

while for pressure

$$p = \frac{p'}{\rho_\infty' U_\infty'^2} \tag{A-3}$$

is used. Here, the subscript ∞ refers to free stream conditions while the primes denote dimensional quantities. In the computer program the number 0 (zero) is used to designate free stream conditions and is interchangeable with ∞. As long as Equation (A-1) through (A-3) are satisfied any system of units may be employed. A complete listing of the required inputs and associated formats is given in Paragraph A. 5.

It should again be mentioned here that this method of starting the three-dimensional characteristics solution from a plane of axisymmetric inputs, is valid only when the geometry downstream of the sphere is such that the sonic point is located on the sphere, for the geometries and angles of attack being considered.

A. 2. 1 Description of Input Data

Input to the program is provided on data cards if the problem to be computed is a new case. If the problem is a restart of a previous incomplete run, input data are provided on cards and on a binary tape. A description of the format and order of the input data cards follows is shown in Table A-1. The input formats have been simplified as much as possible, with 1615 being used for fixed point data and 5E15. 7 being used for floating point as they are consistent with the formulae in Paragraph A. 2. The steps of this input procedure are keyed to the blanks in the accompanying INPUT FORMAT (See Figure A-3). The sign of the exponent is located by a small s. Enter decimal points and right-adjust integer numbers.

TABLE A-I

TABLE OF INPUT DATA CARDS

Card Number	Column Number	MNEMONIC	Item Entered	Number Representation
1	1 - 5	NRUN	Control labeling run number. Any integer number may be used.	Integer I5
2**	1 - 5	ITAPE	Control constant - If ITAPE is not equal to zero, data store is on binary tape.	Integer I5
	6 - 10	NPLIN	Control constant - When ITAPE $\neq$ 0, NPLIN defines the number of the data plane from which the computations are to be re-started.	Integer I5
	11 - 15	IREC4N	Control constant - defines number of records to be skipped on tape 4 when ITAPE $\neq$ 0.	Integer I5
3	1 - 5	NSPEC	Number of species included in gas model description $2 \leq$ NSPEC ≤ 8	Integer I5
	6 - 10	NR	Not used in frozen program - Number of reactions included in non-equilibrium option	Integer I5
	11 - 15	NC	Number of elements	Integer I5

**Note: If ITAPE $\neq$ 0, the only remaining data cards are 3*through 6* below; otherwise cards 3 through 6* are used.

Table A-I (Cont.)

TABLE OF INPUT DATA CARDS

Card Number	Column Number	MNEMONIC	Item Entered	Number Representation
	16 - 20	NVIB	Number of vibrators not in vibrational equilibrium $0 \leq NVIB \leq 2$	Integer I5
	21 - 25	MODEL	0 - frozen solution is computed 1 - ideal gas model is assumed	Integer I5
	26 - 30	NP180	0 - 90° section of flow field is computed 1 - 180° section is computed	Integer I5
	31 - 35	IPTS	Number of points on initial data input ray $3 \leq IPTS \leq 21$	Integer I5
	36 - 40	NPTS	Number of rays in initial data plane $3 \leq NPTS \leq 29$	Integer I5
	41 - 45	NROTAT	Control constant - Used when initial coordinate system is to be rotated to body fixed system	Integer I5
	46 - 50	MDELAY	Control constant - Defines number of rings which are used in surface fitting procedure. Set MDELAY = 3 always.	Integer I5

Table A-I (Cont.)

TABLE OF INPUT DATA CARDS

Card Number	Column Number	MNEMONIC	Item Entered	Number Representation
4	1 - 5 for J = 1 6 - 10 for J = 2 11 - 15 for J = 3 16 - 20 for J = 4 21 - 25 for J = 5 26 - 30 for J = 6 31 - 35 for J = 7 36 - 40 for J = 8	NSPEC(J)	Define which species are used in model description. Species are tagged as follows NOSPEC = 1 for O NOSPEC = 2 for N NOSPEC = 3 for e^- NOSPEC = 4 for Ar NOSPEC = 5 for O_2 NOSPEC = 6 for N_2 NOSPEC = 7 for NO NOSPEC = 8 for NO^+	Integer I5
5	1 - 15	SIGMA	2-D shock angle at shock point on initial ray (degrees)	Floating E15.7
	16 - 30	SIGMA1	2-D shock angle at a shock point slightly downstream of initial ray	Floating E15.7
6	1 - 15	RH0	Free stream density- (gms/cm^3) or ($slugs/ft^3$)	Floating E15.7
	16 - 30	U0	Free stream velocity (cm/sec) or (ft/sec)	Floating E15.7
	31 - 45	R0	Universal gas constant (ergs/mole °K) or (ft lb/mole °R)	Floating E15.7
	46 - 60	T0	Free stream temperature (°K) or (°R)	Floating E15.7
	61 - 75	WMW0	Free stream molecular weight ($\frac{gr}{mole}$ or $\frac{lbs}{mole}$)	Floating E15.7

Table A-1 (Cont.)

TABLE OF INPUT DATA CARDS

Card Number	Column Number	MNEMONIC	Item Entered	Number Representation
7	1 - 15	DIML	Reference length (usually bow shock radius) (cm) or (ft)	Floating E15.7
	16 - 30	HINF	Total enthalpy (nondimensional) $H_\infty = H'_\infty / R'T'_\infty$	Floating E15.7
	31 - 45	X0	x-coordinate of initial plane - Note: x, y, z system is referenced at center of spherical nose (nondimensional) $x_0 = x'_0 / R'_{SN}$	Floating E15.7
	46 - 60	THETFS	Free stream flow angle θ_{fs} (degrees)	Floating E15.7
	61 - 75	PSIFS	Free stream flow angle ψ_{fs} (degrees) (Note: always enter 0.0 here)	Floating E15.7
8	1 - 15	BIGAM	Free stream specific heat ratio	Floating E15.7
	16 - 30	EPSINF (1)	Free stream vibrational energy for first vibrator (Note: If NVIB = 0, EPSINF (1) = 0.0) (nondimensional) $\epsilon_{\infty_1} = \epsilon'_{\infty_1} / R'T'_\infty$	Floating E15.7
	31 - 45	EPSINF (2)	Free stream vibrational energy for second vibrator (Note: If NVIB = 0 or 1, EPSINF(2) = 0.0) (Nondimensional) $\epsilon_{\infty_2} = \epsilon'_{\infty_2} / R'T'_0$	Floating E15.7

TABLE A-I (Cont.)

TABLE OF INPUT DATA CARDS

Card Number	Column Number	MNEMONIC	Item Entered	Number Representation
	46 - 60	PGRAD	Estimated pressure gradient along streamlines $0.95 \leq PGRAD \leq 1.05$ (Expansion) (Compression)	Floating E15.7
	61 - 75	AXIS	Angle final x-axis makes with body axis after rotation (degrees)	Floating E15.7
9	1 - 15	ATTACK	Angle-of-attack (degrees)	Floating E15.7
	16 - 30	ERROR	Error test stop for internal program iterations (usually error = 1.0×10^{-5})	Floating E15.7
10	1 - 15 for J = 1 16 - 30 for J = 2 31 - 45 for J = 3 46 - 60 for J = 4 61 - 75 for J = 5	GAMAM(J)	Free stream concentration for species (J) in order specified by NOSPEC(J) (see card 4) (Note: If J > NSPEC, GAMAM(J) = 0.0). (Nondimensional) ($\delta'_{j\infty} MW'_{\infty}$ moles j/ORIGINAL MOLE)	Floating E15.7
11	1 - 15 for J = 6 16 - 30 for J = 7 31 - 45 for J = 8	GAMAM(J)	Free stream concentrations for species (J) ($6 \leq J \leq 8$) (see details under card 10)	Floating E15.7
12	1 - 15	ALPHA(I)	2-D flow angle (α) for input data point (I) (radians)	Floating E15.7
	16 - 30	R(I)	2-D radius to point (I) on initial line (nondimensional) $r_I = r'_I / DIML$	Floating E15.7

TABLE A-I (Cont.)

TABLE OF INPUT DATA CARDS

Card Number	Column Number	MNEMONIC	Item Entered	Number Representation
	31 - 45	P(I)	Pressure for point (I) on initial line. (nondimensional) $P_I = P'_I / \rho'_\infty U'^2_\infty$	Floating E15.7
	46 -60	Q(I)	Velocity for point (I) on initial line (nondimensional) $Q_I = Q'_I / U'_\infty$	Floating E15.7
	61 - 75	T(I)	Temperature for point (I) on initial line (nondimensional) $T_I = T'_I / T'_\infty$	Floating E15.7
13	1 - 15	RHO(I)	Density for initial point (nondimensional) $\rho_I = \rho'_I / \rho'_\infty$	Floating E15.7
	16 - 30	XMW (I)	Molecular weight for initial point (I) (nondimensional) $MW_I = MW'_I / MW'_\infty$	Floating E15.7
14	1 - 15	EPSI(I, 1)	Vibrational energy of first vibrator for point I. (nondimensional) $\epsilon_{1I} = \epsilon'_{1I} / R'T'_\infty$ (EPSI(I, 1) = 0.0 if NVIB = 0)	Floating E15.7
	16 - 30	EPSI(I, 2)	Vibrational energy of second vibrator for point (I). (nondimensional) $\epsilon_{2I} = \epsilon'_{2I} / R'T'_\infty$ (EPSI(I, 2) = 0.0 if NVIB $\leq$ 1)	Floating E15.7

TABLE A-I (Cont.)

TABLE OF INPUT DATA CARDS

Card Number	Column Number	MNEMONIC	Item Entered	Number Representation
15	1 - 15, J = 1 16 - 30, J = 2 31 - 45, J = 3 46 - 60, J = 4 61 - 75, J = 5	GAMMAI(I, J)	Concentration of species (J) for point (I) on initial data line. (nondimensional) $\gamma_{J_I} = \gamma'_{J_I} MW'_\infty \frac{\text{moles } j}{\text{original mole mix}}$ (if J NSPEC = 0.0)	Floating E15.7
16	1 - 15, J = 6 16 - 30, J = 7 31 - 45, J = 8	GAMMAI(I, J)	See description on card 15	Floating E15.7

Note: Cards 12 through 16 are repeated for each point on the initial line from the body (I = 1) to the shock (I = IPTS) until a total of 11 + 5IPTS cards have been added. This completes the specification of the initial ray input data.

12 + 5IPTS	1 - 15, N = 2 16 - 30, N = 3 31 - 45, N = 4 46 - 60, N = 5 61 - 75, N = 6	OMEGA(N)	Azimuthal angle of ray (N) in initial plane (degrees) (if N > NMAX, OMEGA(N) = 0.0)	Floating E15.7
13 + 5IPTS	1 - 15, N = 7 16 - 30, N = 8 31 - 45, N = 9 46 - 60, N = 10 61 - 75, N = 11	OMEGA(N)	See card 12 + 5IPTS	Floating E15.7
14 + 5IPTS	1 - 15, N = 12 16 - 30, N = 13 31 - 45, N = 14 46 - 60, N = 15 61 - 75, N = 16	OMEGA(N)	See card 12 + 5IPTS	Floating E15.7
15 + 5IPTS	1 - 15, N = 17 16 - 30, N = 18 31 - 45, N = 19 46 - 60, N = 20 61 - 75, N = 21	OMEGA(N)	See card 12 + 5IPTS	Floating E15.7

TABLE A-I (Cont.)

TABLE OF INPUT DATA CARDS

Card Number	Column Number	MNEMONIC	Item Entered	Number Representation
16 + 5IPTS	1 - 15, N = 22 16 - 30, N = 23 31 - 45, N = 24 46 - 60, N = 25 61 - 75, N = 26	OMEGA(N)	See card 12 + 5IPTS	Floating E15.7
17 + 5IPTS	1 - 15, N = 27 16 - 30, N = 28 31 - 45, N = 29	OMEGA(N)	See card 12 + 5IPTS	Floating E15.7
3*	1 - 5	NPRINT	If NPRINT = 0 initial data plane is not printed out	Integer I5
	6 - 10	MODPLN	A plane of output data is printed out every MODPLN integration steps.	Integer I5
	11 - 15	NPLOUT	A plane of output data is printed out on binary tape 4 every NPLUT integration steps.	Integer I5
	16 - 20	NITER	Maximum number of cycles allowed for internal iteration loops. (usually NITER = 10)	Integer I5
	21 - 25	NPLMAX	Maximum integration plane number to be calculated. If this is a restart of a partially completed case, this number should be equal to number of steps already completed + number of steps desired.	Integer I5

TABLE A-I (Cont.)

TABLE OF INPUT DATA CARDS

Card Number	Column Number	MNEMONIC	Item Entered	Number Representation
	26 - 30	IPTMAX	Maximum number of rings allowed $15 \leq IPTMAX \leq 21$	Integer I5
4*	1 - 15	CSST	CSST controls the addition of a new ring of streamlines at the shock. A value close to 1.0 is customary but this may be increased if less detail is required near the shock.	Floating E15.7
	16 - 30	CONST1	Step size control. The step size is multiplied by CONST1 each time it is computed. Usually $0.6 \leq CONST1 \leq 1.0$	Floating E15.7
	31 - 45	CONST2	Step size control. Limits step size when a new ring of field points are added at shock. Usually CONST2 = 1.5	Floating E15.7
	46 - 60	ERROR	See card 9, columns 16-30	Floating E15.7
	60 - 75	XMAX	Maximum x-coordinate to be computed.	Floating E15.7
5*	1 - 5	ISW1	Control constant for internal dumps. See Section A.3 for subroutines involved.	Integer I5

TABLE A-I (Cont.)

TABLE OF INPUT DATA CARDS

Card Number	Column Number	MNEMONIC	Item Entered	Number Representation
	6 - 10	ISW2	Same as ISW1	Integer I5
	11 - 15	ISW3	Same as ISW1	Integer I5
	16 - 20	ISW4	Same as ISW1	Integer I5
	21 - 25	ISW5	Same as ISW1	Integer I5
	26 - 30	ISW6	Same as ISW1	Integer I5
6*	1 - 15	RBDY	Spherical nose radius in cm or ft	Floating E15.7
	16 - 30	SLOPE1	Slope of conical afterbody for circular cone sphere combination (degrees)	Floating E15.7
	31 - 45	SLOPE2	Slope of expansion side for elliptical cone at 0° angle of attack (degrees)	Floating E15.7
	46 - 60	SLOPE3	Slope of compression side for elliptical cone at 0° angle of attack (degrees)	Floating E15.7
	61 - 75	BDYTYP	Define body type = 1.0 for a spherical nose and right circular cone afterbody = 2.0 for a spherical nose and an elliptical afterbody with major axis along $\bar{z}$-direction = 3.0 for a spherical nose and an elliptical afterbody with minor axis along z-direction	Floating E15.7

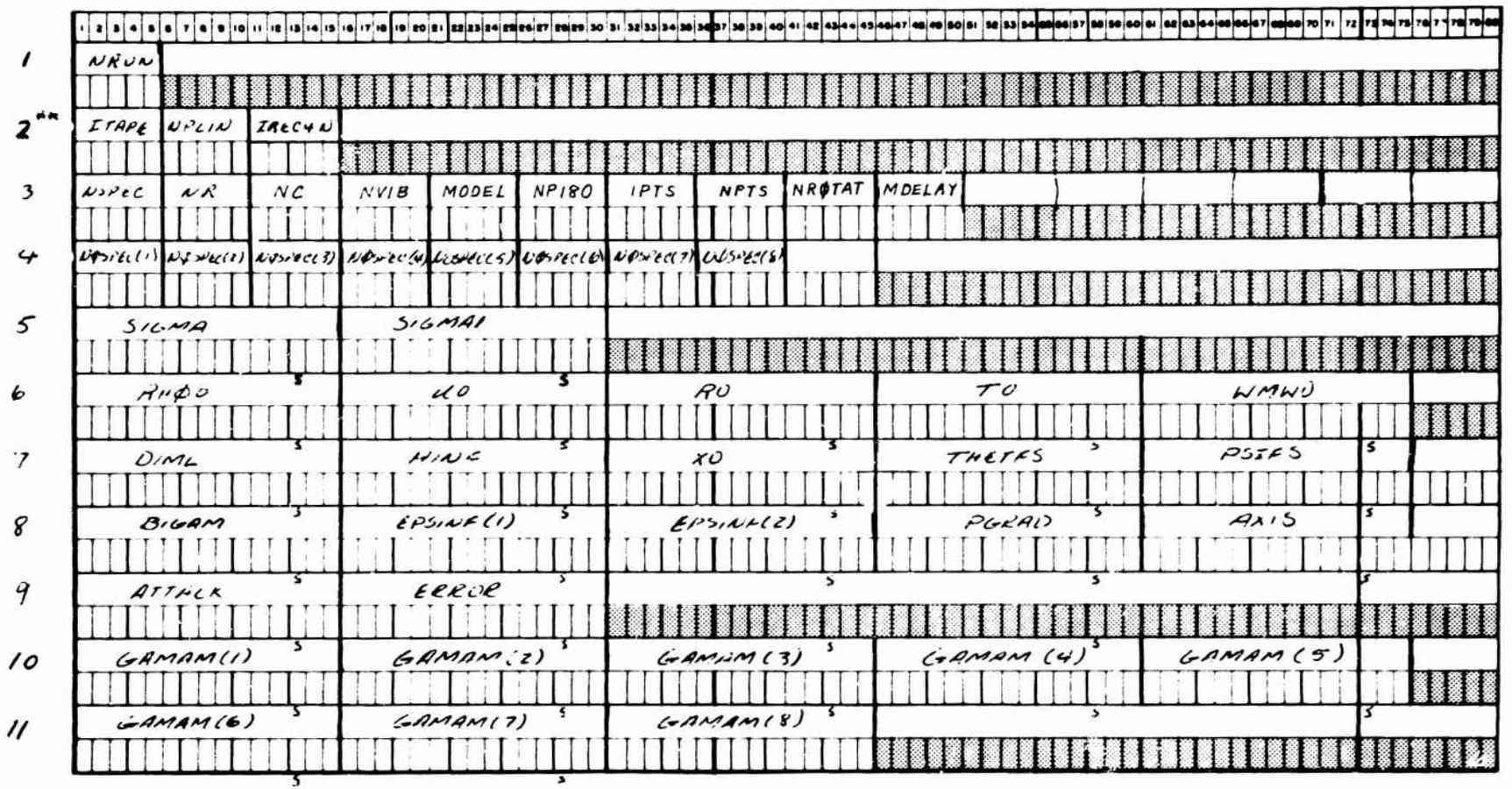


Figure A-3 INPUT FORMAT

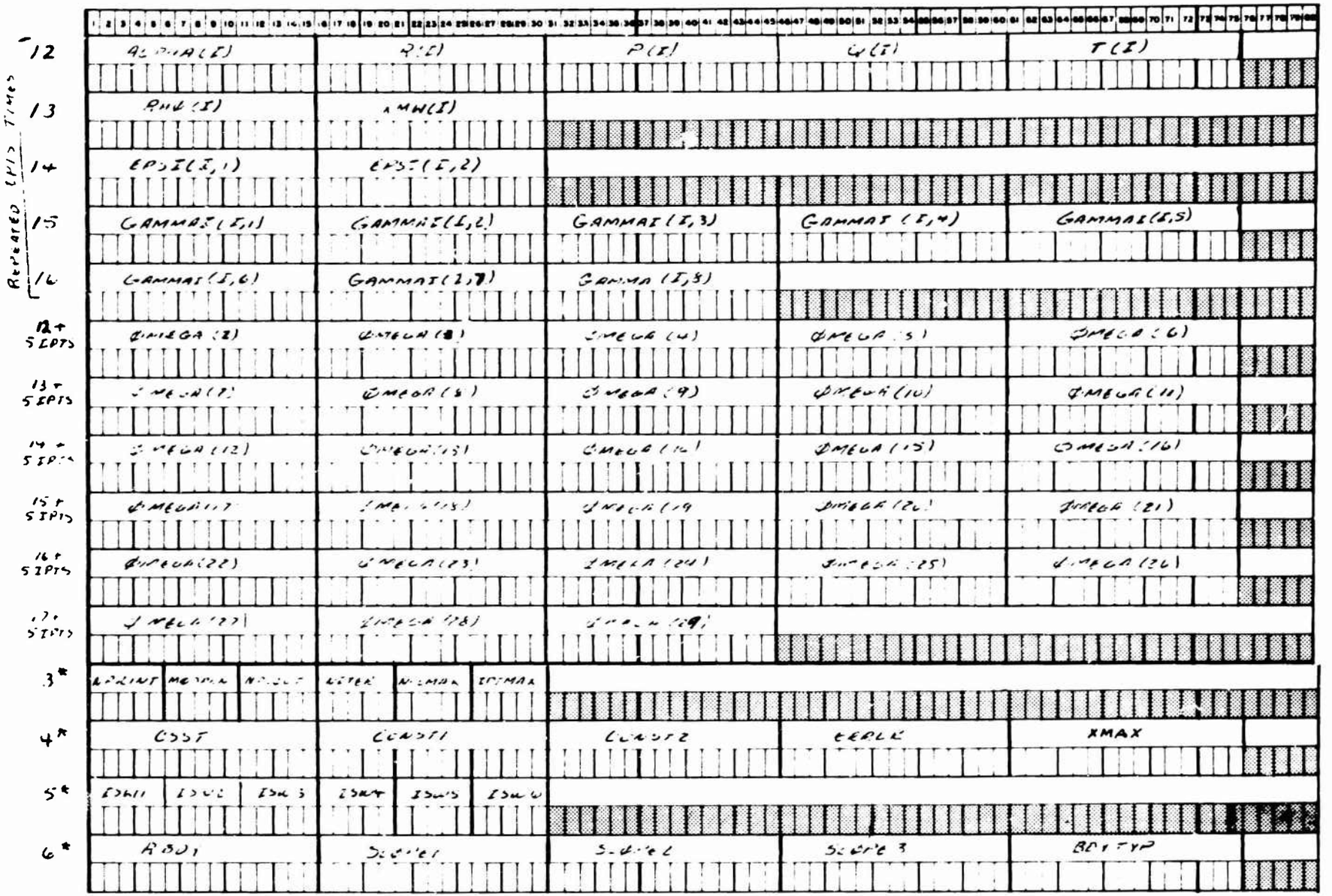


Figure A-3 INPUT FORMAT (Cont)

This completes the description of the input deck. For a new case, a total of 21 + 5IPTS cards are required for input, where IPTS is the number of data points on the initial data ray. For a restart of a previous problem, a total number of 6 data cards are required.

A.3 OUTPUT

The first page of output from the program begins with the run number centered under the heading. Following this come in order: one line of comment (HEAD), the option controls, such as NSPEC, MODEL., etc., the free stream conditions, the free stream compositions and the input body description data. This first page is included initially, and also if the present run is a restart of a previously incomplete case.

Following the initial page of output the values of the point data for a plane of output is usually printed next. The plane printed is the initial plane if the input control constant N PRINT = 1. Otherwise, every MODPLN planes are printed out, where MODPLN is an input constant. The output is arranged by rings starting at the body and ending with the points on the shock surface in the present plane. The form of the output for each point in the plane is shown in Table A-II. The printout for each plane is preceded by a heading (written by subroutine OUTPUT) labeling the form of the following data. Note that the points with the same index I are said to lie on a "ring" and points with the same index N are said to lie on a "ray". Although these were circular rings and straight line rays in the initial plane, they distort somewhat as the computations proceed downstream. However the names ring and ray are maintained for simplicity here.

Other output can be obtained if any or all of the input control constants ISW1, ISW2,___, ISW6 are equal to 1. In this case internal output statements print out certain data within the various subroutines, These are helpful in determining the cause of any problems which may develop during a run. While the forms of these optional outputs will not be given here, they are easily obtained by reference to the appropriate FORTRAN listing. Each optional output is preceded by the subroutine name of the subroutine from which it was written. Following is a list of the subroutines which have internal dumps and the control constants which

initiate their use. Caution should be exercised when using the internal dumps as they generate large amounts of output in a comparatively short time.

1.	ISW1 = 1	Controls	MCMAIN, ROTATE
2.	ISW2 = 1	Controls	POINT
3.	ISW3 = 1	Controls	FIELD, SECT, POINT B
4.	ISW4 = 1	Controls	CDELTA. FLOW, SETUP, SHOCK
5.	ISW5 = 1	Controls	(not used)
6.	ISW6 = 1	Controls	BODY, SECT B

Table A-II
FORM OF OUTPUT FOR EACH PLANE

RESULTS FOR N PLANE = (N PLANE)

		COLUMN NUMBER							
		1	2	3	4	5	6	7	8
ROW NO.	1	RING NO.	RAY NO.						
	2	X	Y	Z	R	OMEGA (DEGREES)	THETA (DEGREES)	PSI (DEGREES)	MOLECULAR WEIGHT
	3	PRESSURE	VELOCITY	DENSITY	TEMPERATURE	SPECIFIC HEAT RATIO	TOTAL ENTHALPY	MACH ANGLE (DEGREES)	MACH NUMBER
	4*	ALPHA 1	ALPHA 2	ALPHA 3					

*NOTE: LINE 4 APPEARS ONLY WHEN THE POINT BEING OUTPUT IS ON THE SHOCK (i.e. THE MAXIMUM RING NUMBER)

A.4 DATA STORE

The data for a plane of points are stored in a large data matrix A1 (25, 30, 15) which allows for up to 25 rings of points with 30 points (rays) per ring and 15 functions per point. For program simplicity, successive blocks of 7500 storage locations have been equivalenced to a named function which in turn is used throughout the program in place of the large matrix A1 whenever a certain one of the functions stored in A1 is required. Following is a list of these new MNEMONICS giving their position in A1 and their significance

Function	Nmemonic		Starting (Position in A1)	Definition
1	VY(25, 30)	=	A1(1, 1, 1)	Y-coordinate
2	VZ(25, 30)	=	A1(1, 1, 2)	Z-coordinate
3	VP(25, 30)	=	A1(1, 1, 3)	pressure
4	VTHETA(25, 30)	=	A1(1, 1, 4)	flow angle, See Fig. 1
5	VPSI(25, 30)	=	A1(1, 1, 5)	flow angle, See Fig. 1
6	VQ(25, 30)	=	A1(1, 1, 6)	velocity
7	VBETA(25, 30)	=	A1(1, 1, 7)	Mach angle
8	VF(25, 30)	=	A1(1, 1, 8)	not used for frozen solution
9	VGAMMA(25, 30)	=	A1(1, 1, 9)	specific heat ratio
10	VMW(25, 30)	=	A1(1, 1, 10)	Molecular Weight
11	VRHO(25, 30)	=	A1(1, 1, 11)	Density
12	VDSTL(25, 30)	=	A1(1, 1, 12)	distance along streamline
13	VX(25, 30)	=	A1(1, 1, 13)	X-coordinate
14	VR(25, 30)	=	A1(1, 1, 14)	Polar RADIUS
15	VOMEGA(25, 30)	=	A1(1, 1, 15)	Polar Angle

A.4.1 Program Stops

When an error (other than Fortran) occurs within the program the program stops with an appropriate error message and full octal core dump. The error message refers back to the statement number of the statement and the subroutine in which the error occurs. In most cases the error is self explanatory.

A. 5 DISCUSSION OF INDIVIDUAL SUBROUTINES

A detailed synopsis of the main program and the individual subroutines is presented in the present section in order to point out techniques and procedures which have not previously been discussed elsewhere in the report.

A. 5. 1 MCMAIN

The main program controls the overall logic of the whole integration scheme, by first locating planes of symmetry and then shifting from one point to another throughout the initial data plane in order to compute new data points in the next plane. The data is arranged so that N data points lie along I rings from the body to the shock, N being constant for all I. Starting on the expansion side of the body at a point located on the shock (the Ith ring) in the plane of symmetry ($N = 1$) N new shock points in the next data plane are successively computed. Control is then shifted to the ring of body points. After N new body points are computed, rings of field points between the body and the shock are computed sequentially. As the calculations progress, data points on the new data plane are written over those in the initial plane which are no longer needed. In this way, as much storage space as possible is saved, since no more than one surface need be stored at a time.

When the ring of shock points is again reached, two options are possible. If a control parameter called IADD is not 0, a new ring of field points corresponding to the streamlines through the shock points is added in the new data plane, otherwise the computation of new points is stopped. In this way, as the shock layer grows along the body, the mesh size through the layer can be maintained fairly uniform without resorting to interpolation procedures in order to add new points to the field. Figure 27 shows some typical streamlines that would be added at the shock in a plane of symmetry. If a new ring of field points is added at the shock, the ring of shock points in the new data plane is indexed up to $I + 1$ and this data plane then becomes a new initial plane from which another plane can be calculated. New planes are calculated in this manner, until either a maximum number of planes have been determined, or a maximum x coordinate is reached.

The main program also has the option of dropping whole rings of data points whenever desired. Presently this is accomplished by first testing whether the total number of rings I is equal to the maximum number allowable IPTMAX, an input constant to the program. If I equals IPTMAX, and the mesh spacing near the shock is such that a new ring of shock points would be added in the next data plane (again in order to keep mesh spacings fairly uniform) then, every other ring of field points between the body and the shock is dropped from the computation. This technique takes advantage of the fact that since the calculations proceed along the x-axis and the flow gradients decrease, less detail is required to completely specify the flow field. Hence, the overall computation procedure can be drastically speeded up -- both from the points of view of having now fewer points to calculate, and having also larger mesh widths in the initial data plane, which in turn allow for larger axial step sizes Δx. For many problems it may be desirable to drop rings other than these throughout the flow field. This is especially true for some bodies at high angles-of-attack when the spacings between rings near the body on the compression side may tend to be small. By checking spacings between the rings in say the plane of symmetry on the compression side of the body, any specified rings can be dropped from the procedure.

MCMAIN calls subroutines OUTPUT, STOP, STEP, FIT, BODY, FIELD and SHOCK in that order. The function of each of these routines is given below. Since none of the actual computations are performed in MCMAIN, only slight changes to account for any extra functions which are added will be required in order to include different thermochemical models in the program.

A.5.2 INPUT

This subroutine reads all the input data that is required by the program from a tape where it had previously been written in the proper order and format. INPUT has essentially two separate functions, depending upon what data is read from the tape. If the data is that obtained from a blunt body solution along one line only, it is first nondimensionalized using the methods given in Paragraph A.3, and then transferred to several different azimuthal planes using the fact that the initial data should be axisymmetric. In this case, the input data is also written on BCD output tape for checking purposes.

If the data is that read from a binary tape (auxiliary tape 2) which has previously been written by the program (see subroutine STOP) it then represents a continuation of a previously run problem and is already in the form necessary for use by the program. Hence, control is transferred back to MCMAIN immediately.

The initial point data thus acquired by INPUT is stored in a common three-dimensional array labeled A1. Any change in the thermochemical model would be reflected in a corresponding increase in the size of A1 in storage. The correspondence between the elements of A1 and the nomenclature used in the program is given in Paragraph A.4. Calls SMALLH and PAUSE.

A.5.3 STEP

The STEP size Δx between successive data planes normal to the x-axis is a function of the spacing of the points on the base plane, the magnitude and direction of the velocity field, and the speed of sound throughout the field. Here, Δx is chosen so that the Mach forecone from any new point intersects the base plane in such a manner that its domain

of dependence does not contain any initial points adjacent to the original base point. This can in general be assured by taking

$$\Delta x \leq DIST\,(\cos(\beta_{max} + \phi_{min})/\sin\beta_{max}) \qquad (A\text{-}4)$$

where DIST is the minimum mesh point spacing, β_{max} is the maximum Mach angle and ϕ_{max} is the maximum angle which the velocity vector makes with the x-axis. Usually, however, it will not be necessary to check every point in the initial plane in order to satisfy (A-4). In the present version of STEP, only points near the body and the shock in the geometrical planes of symmetry are checked in order to determine Δx. Δx as obtained from (A-4) is then multiplied by an input constant (CONST) to yield the final value of Δx. For problems where the maximum values of the Mach angle and flow angle are expected off the planes of symmetry (e.g., an ellipsoidal body at angle of attack), CONST1 can be appropriately adjusted so that the computation can proceed.

Another function of STEP is to test the spacing of data points near the shock surface to determine whether or not a new streamline will be added at the shock. If new streamlines are required, the test constant IADD (see MAIN) is set equal to one -- otherwise, IADD equals zero.

A.5.4 FIT And LSTSQ

Interpolating surface fits through nine initial data points are obtained in this subroutine along the lines discussed in Step 1 of Paragraph 3.3.2. The functions fit are the pressure, velocity and two flow angles. Since the flow angles θ and ψ vary considerably around a given body (e.g., at any x-location for an axisymmetric flow, θ varies from a maximum along the y-axis, to zero along the z-axis) a transformation to new spherical angles ω, $\varkappa$ is first performed. The justification for for this lies somewhat in the fact that for many three-dimensional problems considered, the flow angle ω will vary mainly with one of the independent variables, while $\varkappa$ will vary mainly with the other. The relationship between the angles ω, $\varkappa$ and the angles θ, ψ is shown in Figure (A-4).

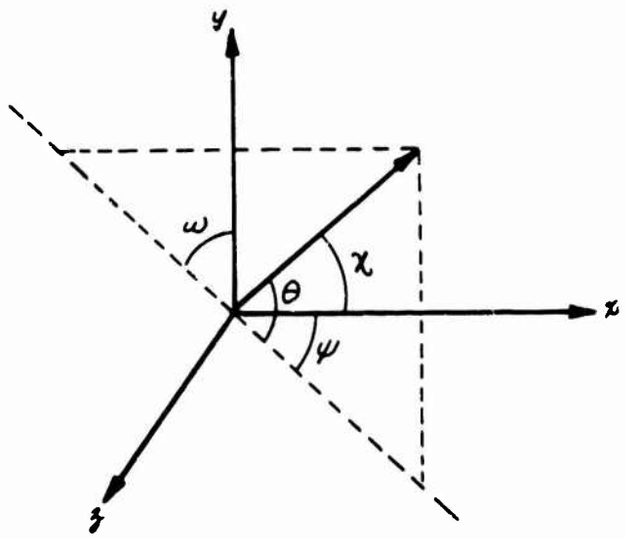


Figure A-4 θ, ψ TO ω, χ COORDINATE TRANSFORMATION

The independent variables presently used in the fitting technique are the polar coordinates (r , Ω) in the yz-plane. Here,

$$\Omega = \tan^{-1}\left(\frac{z}{y}\right) \tag{A-5}$$

is the angle between the y-axis and the point. While these coordinates are generally best for most "rounded" bodies, (e.g., axisymmetric or ellipses of moderate eccentricities), for bodies with locally two dimensional regions (e.g., a "flat" ellipse) perhaps a combination of cartesian coordinates (y, z) over the flat portion and polar coordinates (r, Ω) over the rounded portions would be better. In this case, the focal point of the ellipse would be chosen as the center of the (y, z) coordinate system. This can better be determined after gaining more experience with the program. Call HETA, SIMSOL.

A.5.5 FIELD (IP, NP)

Given an initial field point (IP, NP), the step size Δx and the results from subroutine FIT, the FIELD subroutine calculates the coordinates and flow properties at a new field point. To do this, subroutine POINT, STREAM, SECT, INTPLT and SETUP are used throughout an iteration loop to converge the solution to second order in the step size, along the lines proposed in Paragraph 3.3.2.

As an error check in the modified Euler iteration scheme, a limit of NITER cycles is set on the looping. If the convergence test on pressure

$$\frac{p^{(n)} - p^{(n-1)}}{p^{(n)}} < 0.00001 \tag{A-6}$$

is not satisfied, within these NITER cycles, an error statement is written off-line and control is returned to the main program. Calls SIMSOL, INTPLT, POINT, STREAM, POINTB, SCKFN, SHIFT, SECT, SETUP.

A.5.6 BODY (IP, NP)

Given an initial body point (IP, NP), the BODY subroutine calculates the flow properties at a point where the body streamline through (IP, NP) meets the next data plane. To accomplish this, BODY calls exactly the same subroutines as FIELD plus two more given the mnemonics BODYFN and SECTB. The procedure is essentially that given in Paragraph

3.3.3, where now, two iteration loops are required in order to converge the solution to second order in the step size.

The inner loop uses Newton iteration to converge the flow angles θ and ψ such that the boundary condition of flow tangency at the surface is solved with the absolute limits

$$\theta^{(n)} - \theta^{(n-1)} < 10^{-8} \tag{A-7}$$

and

$$\psi^{(n)} - \psi^{(n-1)} < 10^{-8} \tag{A-8}$$

If either of the Equations (A-7) and (A-8) is not satisfied within 20 cycles an error statement is again printed off-line and the computations allowed to continue.

The outer iteration is again a modified Euler type on pressure and contains the same error checks and convergence criterion as does subroutine FIELD.

A. 5. 7 SHOCK (IP, NP)

Given an initial shock point (IP, NP), this subroutine calculates the coordinates, flow properties and direction cosines of the shock normal at a new point on the shock surface in the next data plane. The method is that given in Paragraph 3. 3. 4. In the process, SHOCK uses subroutines FLOW, CDELTA and SETUP, whose various functions are explained below, to converge the solution to again, second order in the step size. Here again, the convergence tests on pressure are the same as those used in FIELD and BODY with the appropriate error statement included. Calls SHIFT, PAUSE, POINT.

A. 5. 8 INTPLT

Given the cartesian coordinates (y, z) of a point P in the base plane, INTPLT calculates its polar coordinates (r, Ω) (See Equation (A-4)) and uses these in the equations defined by FIT in order to determine pressure, velocity and flow angles θ and ψ at P. θ and ψ are obtained from ω and $\varkappa$ (the actual angles fit) by using the inverse transformations (See Figure (A-4).

$$\psi = \tan^{-1}(\tan\alpha \sin\omega) \tag{A-9}$$

and

$$\theta = \tan^{-1}(\cot\omega \sin\psi) \tag{A-10}$$

CALLS HETA

A.5.9 STREAM

The compatibility equations written along the streamline are solved in STREAM using in this case, modified Euler integration. For equilibrium and frozen flow, this same integration technique can be used. However, for nonequilibrium flow a more accurate technique using the methods discussed in Paragraph 3.3.2 would require an entirely new STREAM routine. Calls PAUSE.

A.5.10 SECT

Given the flow properties at the new point and the bicharacteristic parametric angle δ, this subroutine solves for the coordinates of the new base point in the initial plane in the manner indicated in Step 3 of Paragraph 3.3.2.

A.5.11 SETUP

In this subroutine the coefficients of the compatibility equations written along the bicharacteristics are obtained in the manner indicated in Step 5 of Paragraph 3.3.2. For nonequilibrium flow, the only change required here would be to add the term $-\sin\beta_i \bar{F}_i \Delta L_i$ to the calculations along the bicharacteristic i. Calls HETA.

A.5.12 SECTB

In SECTB, the new body point is located as the intersection of the plane containing the normal to the body and the streamline through the initial body point with the equation of the body surface and the next plane. Since the equation of the body surface usually changes for every problem, the equations are solved using a second order Newton process. Calls BDYFN, PAUSE.

A.5.13 BODYFN

The equation of the body

$$B(x, y, z) = 0$$

and the direction numbers of the body normals are supplied by BODYFN, which must be rewritten if a sphere cone is not desired. Here the conical afterbody may be circular or elliptical as specified on input. The body functions are written with respect to a body centered coordinate system with the x-axis as the body axis. Eventually, BODYFN should be replaced by a general routine which allows for either an analytical description or a numerical surface fitting technique to describe the body..

A.5.14 SCKFN

Three adjacent shock points on the shock surface in the initial plane are fit with an exact quadratic fit in the polar coordinates (r, Ω) by SCKFN. Calls HETA.

A.5.15 CDELTA

This subroutine locates three bicharacteristics passing back from the new shock point to the initial plane, using the method presented in Step 4 of Paragraph 3.3.4. A Newton-Raphsen iteration scheme is used to solve for the base coordinates of each of two bicharacteristics imbeded in the shock surface. Then, a third bicharacteristic whose direction is represented by the average of these two is located. CDELTA also calls INTPLT in order to determine the base point flow properties. Calls SCKFN, POINTB, SIMSOL, HETA, INTPLT.

A.5.16 STOP

Subroutine STOP has two basic functions. First, a logical binary record containing all the data which would be necessary to restart the computations is written for every N planes, where N is an input constant to the program. Then, if either a given maximum number of planes have been computed, or a given maximum x-coordinate has been reached, the computational procedure is halted, otherwise, control is transferred back to MCMAIN.

A.5.17 OUTPUT

Output data from the program is written for every N planes, where here, N is the same input constant which was used in STOP. OUTPUT can easily be changed to provide any output quantity desired, without affecting the reaminder of the program. The form of the output obtained at each point in the plane in the present version of OUTPUT, is described in Paragraph A.3. OUTPUT calls POINT which prints out the point data for each point in the plane.

A. 5. 18 SIMSOL

Any set of n linear inhomogeneous equations in n unknowns can be solved in SIMSOL. If the determinant of the coefficients vanishes, an appropriate error statement is printed out.

A. 5. 19 DIVIDE

DIVIDE is used to drop every other ring of data points from the shock to the body.

A. 5. 20 ROTATE

ROTATE is used to rotate the coordinates such that all the data points lie in a plane x = constant. The option which utilizes this subroutine is not presently working properly. Calls HETA.

A. 5. 21 POINT

POINT is used to write out the data at a given data point. The form of this output is given in Paragraph A. 3.

A. 5. 22 POINTB

POINTB is used to provide extra output data at the base of each bicharacteristic. It is utilized mainly for debugging purposes.

A. 5. 23 PAUSE

If an error occurs which would make it impossible to continue the procedure, an appropriate error statement is written by PAUSE, and the program is terminated.

A. 5. 24 HETA

Determine an angle θ , in radians ($-\frac{\pi}{2} \leq \theta \leq \frac{\pi}{2}$) when given the opposite and adjacent sides of a right triangle.

A. 5. 25 FLOW

FLOW is used to compute the conditions behind an oblique shock given the shock angle, σ , and the free-stream conditions at each point. For all thermodynamic options the computations utilize a Newton-Raphson procedure to match the total enthalpy behind the shock with that of the free stream. FLOW calls SMALLH and PAUSE.

A. 5. 26 SMALLH

SMALLH is presently used to provide harmonic oscillator approximations to the species thermodynamic functions enthalpy, h_j , specific heat, C_{pj} , and free energy μ_j . The harmonic oscillator data is contained within the program in the form of data statements with a maximum of 8 species allowed. These are presently, O, N, E^-, A, O_2, N_2, NO, NO^+, in that order. One data card is read in INPUT in order to choose the proper data for the thermochemical system being run. A description of this card is included in the section on the discussion of the program input data.

A. 5. 27 STMN

Control program which is used to read run label and control the overall execution of the program. STMN calls INPUT and MCMAIN.

A. 5. 28 SHIFT

SHIFT is used to shift the point data from location I1 to location I2.

A. 5. 29 COMMON DECK

Must be used with all subroutines except SIMSOL, HETA and PAUSE.

A.6 SYMBOL DEFINITION

Following is a list of symbols which are used throughout the program. Arrangement is in the order the variable appears for each labeled common block followed by the list of special variables used within each subroutine. The asterisk designates input quantities.

Mnemonic Variation	Symbol	Comment
COMMN/FREE		
*P0 = PRES	p'_∞	Free-stream pressure in dynes/cm^2 or lb/ft^2
*RHO0 = DENS	ρ'_∞	Free-stream density in grams/cm^3 or slugs/ft^3
*U0 = VEL	U'_∞	Free-stream velocity in cm/sec or ft/sec
*WMW0 = WGHT	MW'_∞	Free-stream molecular weight in $\frac{\text{grams}}{\text{mole}}$ or $\frac{\text{lbs.}}{\text{mole}}$
*T0 = TEMP	T'_∞	Free-stream temperature in °K
*HINF	H_∞	Free-stream total enthalpy normalized by $\bar{R}'T'_\infty/MW'_\infty$
TINF	T_∞	Free-stream temperature normalized by T'_∞
PAM	P_∞	Free-stream pressure normalized by $\rho'_\infty U'^2_\infty$
RHAM	ρ_∞	Free-stream density normalized by ρ'_∞
UAM	U_∞	Free-stream velocity normalized by U'_∞
WTAM	MW_∞	Free-stream molecular weight normalized by MW'_∞
*DIML		Reference length in cm or ft
*R0	$\bar{R}'$	Universal Gas constant $\frac{\text{ergs}}{\text{mole-°K}}$ or $\frac{\text{ft.lb}}{\text{mole-°R}}$
BIGAM	γ_∞	Free-stream specific heat ratio

Mnemonic Variation	Symbol	Comment
XLMBDA - TSCALE	Λ	Defined as $\frac{MW'_\infty U'^2_\infty}{\bar{R}' T'_\infty}$
*EPSAM(K) =EPSINF(K)	ϵ_{∞_k}	Free-stream vibrational energy for kth vibrator normalized by $\bar{R}'T'_\infty/MW'_\infty$
*GAMAM(J) =GAMINF(J)	γ_{∞_j}	Free-stream concentration of Jth species normalized by $\sum_j \gamma_{\infty_j}$
COMMON/THERMO		
HJ0(J)	h_{j_o}	Energy of formation of jth species nondimensionalized by $\bar{R}'T'_\infty/MW'_\infty$
H(J)	h_j	Enthalpy of jth species nondimensionalized by $\bar{R}'T'_\infty/MW'_\infty$
FJ(J)	f_j	Free-energy of jth species nondimensionalized by $\bar{R}'T'_\infty/MW'_\infty$
CP(J)	c_{p_j}	Specific heat of jth species nondimensionalized by $\bar{R}'/MW'_\infty$
EPSEQ(K)	ϵ_{∞_k}	Equilibrium vibrational energy of kth vibrator normalized by $\bar{R}'T'_\infty/MW'_\infty$
NATOMS(J)	N_j	Number of atoms in species j.
COMMON/DATA3D		
A1(I,J,K)		Storage matrix for point data I = ring number, J = ray number, K = variable number
ALF2(N)		Direction cosine of shock normal for shock point (N) along y-axis
ALF3(N)		Direction cosine of shock normal for shock point (N) along z-axis
ALFL2(N)		ALF2(N) for previous shock point (N)
ALFL3(N)		ALF3(N) for previous shock point (N)
*GAMMAI (I, J)		Concentration for ring I and species J
*EPSI (I, K)		Vibrational energy for ring I and vibrator K

Mnemonic Variation	Symbol	Comment
COMMON/BOSS3D		
AM(9, 10)		Matrix used for storing coefficients of linear equations to be solved simultaneously
ALPHAL		Angle between present x-axis and body axis when plane rotation is used
ALPHAN		Angle between normal to rotated plane and body axis
BETAMX		Maximum Mach angle
CCP		Specific heat ratio (nondimensional)
CSIGMA		Cosine of shock angle
SSIGMA		Sine of shock angle
CPSIFS		Cosine of free stream flow angle
SPSIFS		Sine of free stream flow angle
CTHEFS		Cosine of free stream flow angle
STHEFS		Sine of free stream flow angle
*CONST1		Input constant used to regulate step size,
*CONST2		Input constant used to decrease step step size when new field points are added at the shock
DALPHA		Angle of rotation of data plane when coordinate rotation is employed
DLMAX		Not used
DXMIN		Minimum step size allowed
*IPTS		Number of data rings for present step
ICONST(10)		Integer matrix (not used)
*IPTMAX		Maximum number of data rings allowed

Mnemonic Variation	Symbol	Comment
*ISW1 *ISW2 *ISW3 *ISW4 *ISW5 *ISW6		Integer test constants used to control internal program print out
*ITAPE		Integer Control constant (see input)
*MDELAY		Integer constant always set = 3 on input
*MODPLN		See description in input section
*MODEL		See description in input section
*NC		See description in input section
NEWRUN		Control constant used to test whether or not a subroutine has been previously entered
*NP180		See description in input section
*NPCMAX		See description in input section
*NPLOUT		See description in input section
*NPRINT		See description in input section
*NROTAT		See description in input section
*NPTS		See description in input section
NPTMAX		Not used
NPLANE		Number of current integration plane
*NR		See description in input section

Mnemonic Variation	Symbol	Comment
NRCORD		Not used
*NVIB		See description in input section
*NSPEC		See description in input section
*PGRAD		See description in input section
PP		Not used
QQ		Not used
*SIGMA		Shock angle
*SIGMA1		Shock angle for upstream shock point
THETMN		Minimum flow angle
VARIBL(10)		Real matrix (not used)
X0		x-coordinate of present data plane
Y0		y-coordinate of last shock point on present data plane
*XMAX		Maximum x-coordinate calculated
*NOSPEC(N)		See description in input section
COMMON/AFT3D		
A2(J, M, N)		Matrix used in setting up quadratic surface fits.
AMUB(JJ)		Mach angle at bicharacteristic (JJ) base point.
AMUT3		Mach angle at new point
AMUAV		Average Mach angle along bicharacteristic
ALFT1 ALFT2 ALFT3		Temporary storage locations for direction cosines to shock normal

Mnemonic Variation	Symbol	Comment
AT32(K)		Flow angle, θ , as computed by the (K) set of bicharacteristics
AT32C		Average pressure at new shock point or average flow angle, θ , at new field point
AT33(K)		Flow angle, ψ , as computed by the (K) set of bicharacteristics.
AT33C		Average flow angle, ψ , at new point
AT31(K)		Pressure as computed by the (K) set of bicharacteristics
AT31C		Average pressure at new point
AT3T(I,K)		Storage matrix for pressure (I=1) for flow angle θ (I=2) and for flow angle ψ (I=3) as computed from the (K) set of bicharacteristics
AVE(J)		AVE(J) is equal to value of function (J) for middle point of nine points fit.
AMUBT(J,K)		Mach angle for bicharacteristic (J) in the (K) set of bicharacteristics
ALF1	α_1	Direction cosine of normal to shock with respect to x-axis
B0		Body function B(x, y, z) = 0 solved at point (x, y, z) gives remainder B0.
CDEL	$\cos(\delta)$	Cosine of parametric angle δ at new point
SDEL	$\sin(\delta)$	Sine of parametric angle δ at new point
CDELB	$\cos(\delta_0)$	Cosine of parametric angle δ at bicharacteristic base point
SDELB	$\sin(\delta_0)$	Sine of parametric angle δ at bicharacteristic base point

Mnemonic Variation	Symbol	Comment
CAMU	$\cos(\mu)$	Cosine of Mach angle
SAMU	$\sin(\mu)$	Sine of Mach angle
CAMUB	$\cos(\mu_0)$	Cosine of Mach angle at bicharacteristic base point
SAMUB	$\sin(\mu_0)$	Sine of Mach angle at bicharacteristic base point
CTHET	$\cos(\theta)$	Cosine of flow angle θ
STHET	$\sin(\theta)$	Sine of flow angle θ
CTHETB	$\cos\theta)$	Cosine of flow angle θ at bicharacteristic base point
STHETB	$\sin(\theta_0)$	Sine of flow angle θ at bicharacteristic base point
CTHET0	$\cos(\theta_0)$	Cosine of flow angle θ_0
STHET0	$\sin(\theta_0)$	Sine of flow angle θ_0
CPSI0	$\cos(\psi_0)$	Cosine of flow angle ψ_0
SPSI0	$\sin(\psi_0)$	Sine of flow angle ψ_0
C01		Direction number of body slope with respect to x-axis at initial body point
C02		Direction number of body slope with respect to y-axis at initial body point
C03		Direction number of body slope with respect to y-axis at initial body point
CT1 CT2 CT3		Direction number of body slope with respect to x, y- and z-axis respectively for new body point
COEFN(N, J)		Coefficients used in interpolating surface fits for function J.
DELTA(J)	δ_j	Parametric angle δ for bicharacteristic J.

Mnemonic Variation	Symbol	Comments
DELTAB(J)	δ_b	Parametric angle (δ_b) at base point of bicharacteristic J.
DELTBT(J, K)		Parametric angle for bicharacteristic (J) in set of bicharacteristics (K).
DERFN(M, N)		not used
DADX DADY DADZ		Derivative of speed of sound with respect to x, y and z axes respectively
DADN		Derivative of speed of sound with respect to N-direction
DBDXAV DBDYAV DBDZAV		Average derivative of body function with respect to the x, y and z axes respectively
DALF2		Change in direction cosine of shock normal with respect to y-axis.
DALF3		Change in direction cosine of shock normal with respect to z-axis
DELTAV		Average parametric angle along a bicharacteristic
DL(J)		Length of bicharacteristic (J)
DXNEXT		Step size along x-axis between successive data planes
DXDN DYDN DZDN		Partial derivatives of x, y and z coordinates with respect to the N-axis respectively.
E1 E2 E3 F1 F2 F3 G1 G2 G3		Coefficients used to compute $\frac{\partial\theta}{\partial x}, \frac{\partial\theta}{\partial y}, \frac{\partial\theta}{\partial z}, \frac{\partial\psi}{\partial x}, \frac{\partial\psi}{\partial y}$ and $\frac{\partial\psi}{\partial z}$ from simultaneous solution of equations of the form $\theta - \theta_i = \frac{\partial\theta}{\partial x_{ij}} (x_j - x_{ij})$ (see Subroutine SETUP)

Mnemonic Variation	Symbol	Comments
ETLAST		
EPLAST		
F(J)		Basis functions (J) used in interpolating polynomial
FNORM		Normalizing factor for unit normal to body surface at a point on body
FBT(3, 3)	$\mathcal{F}$	not used for frozen solution
FB(3)	$\mathcal{F}_\theta$	not used for frozen solution
FUNCT(J)		Solution for function (J) obtained from interpolation using polynomial surface fits.
IADD		IADD = 1 if new ring of field points is to be added at shock IADD = 0 otherwise
IFIT		Number of rings used around center ring in polynomial fit. IFIT = 1 always
IDP		Number of rings used in surface fitting procedure. IDP = 3 always.
NDP		Number of rays used in surface fitting procedure. NDP = 3 always.
NFUNCT		Number of functions fit. NFUNCT = 8 always
NOPT		Number of points included in polynomial surface fit. NOPT = 9 always
NOPTS		NOPT + 1
NO FN		Constant defining function presently used when a large matrix contains several different functions in sequence (i.e. A1)

Mnemonic Variation	Symbol	Comments
NFIT		Number of rays used around center ray in surface fitting procedure
NN		Iteration counter
NOCHAR		Number of bicharacteristics used in unit mesh procedure
PSIB(J)	ψ_{B_j}	Flow angle at base of bicharacteristic (J).
PSIBT(J, K)		Flow angle ψ at base of bicharacteristic (J) for set of bicharacteristics (K).
PSIAV		Average flow angle ψ along a bicharacteristic
PPP		not used
PLAST		Last value of pressure obtained in iteration procedure
PB(J)	p_{B_j}	Pressure at base point of bicharacteristic (J).
P1		not used
PPB		not used
OMEGAB	ω_B	Polar angle of bicharacteristic base point with respect to y-axis.
Q1		Velocity at initial point
Q232		Not used
QP		Not used
QQB		Velocity at bicharacteristic base point.
QS		Not used
QB(J)	q_{B_j}	Velocity at base point of bicharacteristic(J).

Mnemonic Variation	Symbol	Comments
RB	r_B	Radius from origin to bicharacteristic base point
R01 R02 R03		Direction numbers of line of intersection of plane tangent to shock surface and plane normal to shock surface containing velocity vector behind the shock at the initial shock point.
R1AV R2AV R3AV		Average line in shock surface for locating new shock point defined by R1AV = R01 + RT1 etc.
RT1 RT2 RT3		Direction numbers of line of intersection of plane tangent to shock surface and plane normal to shock surface containing velocity vector behind the shock at the new shock point.
RR		not used
SSB(J)		Speed of sound at base point of bicharacteristic (J)
SST3		Speed of sound at new data point
SSA1		Speed of sound at initial data point
SS		Speed of sound
SCALE(J)		Scaling factor for polynomial fit to function (J).
THETAB(J)	θ_{B_j}	Flow angle, θ , at bicharacteristic (J) base point
THBT(J,K)		Flow angle, θ , at base of bicharacteristic (J) for set of bicharacteristics (K).
TA1		Temperature at initial data point
THETAV	$\bar{\theta}$	Average flow angle, θ , along a bicharacteristic.

Mnemonic Variation	Symbol	Comments
TAT3		Temperature at new data point
UAVE	$\bar{u}$	Average x-component of velocity
VAVE	$\bar{v}$	Average y-component of velocity
WAVE	$\bar{w}$	Average z-component of velocity
US	$u_{(-)}$	x-component of velocity behind shock
VS	$v_{(-)}$	y-component of velocity behind shock
WS	$w_{(-)}$	z-component of velocity behind shock
XB(J)		x-component of base point for bicharacteristic(J).
XP(4)		not used
XPT XNEXT		x-coordinate of new point when plane rotation is not employed
YB(J)	y_{B_j}	y-coordinate of base point for bicharacteristic (J).
YP(4)		not used
YPT		not used
ZB(J)	z_{B_j}	z-coordinate of base point for bicharacteristic (J)
ZP(4)		not used
GAMB(J)		Specific heat ratio at base point bicharacteristic(J).
GAMBT(J,K)		Specific heat ratio at base point of bicharacteristic (J) for set of bicharacteristics (K).

COMMON LOOP

*NITER		Maximum number of iterations allowed

Mnemonic Variation	Symbol	Comments
*ERROR = CHI		Iteration limit - $\sim 1.0 \times 10^{-5}$ usually
COMMON/TTTAPE		
*NREC4		Number of records on binary tape 4.
NREC3		Number of records on binary tape 3.
COMMON/STIPN		
*CSST		Defined in input section
COMMON/BDYDAT		
*RBDY		Spherical cap nose radius in cm or ft.
XBDY1		x-coordinate of beginning of conical afterbody portion of body.
XBDY2		x-coordinate of beginning of expansion side for elliptical afterbody
XBDY3		x-coordinate of beginning of compression side for elliptical afterbody
YBDY1		y-coordinate of beginning of conical afterbody portion of body
YBDY2		y-coordinate of beginning of expansion side for elliptical afterbody
YBDY3		y-coordinate of beginning of compression side for elliptical afterbody
*SLOPE1 *SLOPE2 *SLOPE3 *BDYTYP		Defined in input section
COMMON/OUTPNT/ IPOINP		not used

Mnemonic Variation	Symbol	Comment
SUBROUTINE INPUT		
*ALPHA(I)		2-D flow angle for input point (I) (radians)
BETA(I)		Mach angle for input point (I)
CH(I)		Total enthalpy for input point (I)
CPP(I)		Specific heat ratio for input point (I)
CM		Mach number
G(I)		Not used
GAMEFF		Frozen specific heat ratio
GAMMA(I)		Frozen specific heat ratio for input point (I)
*OMEGA(N)		Azimuthal angle for ray (N)
*P(I)		Pressure for input point (I)
*Q(I)		Velocity for input point (I)
*R(I)		Radius for input point (I)
*RHO(I)		Density for input point (I)
SOUND(I)		Speed of sound for input point (I)
*T(I)		Temperature for input point (I)
SUBROUTINE BODY		
D1 E1 F1		Combination of coefficients of compatibility equations used to eliminate pressure term and obtain one equation in two unknown (θ , ψ)
DELT		Parametric angle, δ , for bicharacteristic in normal plane to body at new point

Mnemonic Variation	Symbol	Comment
DFDPSI	$\frac{\partial F}{\partial \psi}$	Partial derivative of function FN with respect to flow angle ψ
DFDTH	$\frac{\partial F}{\partial \theta}$	Partial derivative of function FN with respect to flow angle θ
DGDPSI	$\frac{\partial G}{\partial \psi}$	Partial deviative of function G with respect to angle ψ
DGDTH	$\frac{\partial G}{\partial \theta}$	Partial derivative of function G with respect to flow angle θ
DENOM		Denominator for Newton-Raphson method in two variables (G and FN) in terms of two independent variables (θ , ψ).
DTHETA	$\Delta\theta$	Change of angle θ.
DPSI	$\Delta\psi$	Change of angle ψ.
FN		Function representing condition that velocity vector is tangent to body
G		Function representing combination of compatibility equations eliminating pressure, p.
SILAST		Previous value of flow angle ψ in iteration
THLAST		Previous value of flow angle θ in iteration
X		X = $\cos\delta$
SUBROUTINE BODYFN		
B		Body function
C1 C2 C3		Direction numbers of outward normal to body in body oriented coordinates
C1NEW C2NEW C3NEW		Direction numbers of outward normal to body in external coordinate system

Mnemonic Variation	Symbol	Comment
CALF	$\cos(\alpha)$	Cosine of angle between body axis and present external x-axis
SALF	$\sin(\alpha)$	Sine of angle between body axis and present external x-axis
FFNORM		Norm of direction numbers
SMAJ		Length of semi-major axis of body ellipse
SMIN		Length of semi-minor axis of body ellipse
X, Y, Z		Coordinates of body point with respect to body oriented Cartesian system
XNEW, YNEW, ZNEW		Coordinates of body point with respect to external coordinate system

SUBROUTINE CDELTA

CFSK1, CFSK2, CFSK3		Coefficients for quadratic fit to shock curve in x = const. plane
D1, D2, D3, D4, D5, D6, D7		Coefficients of Mach forecone from new shock point
DFNDY	$\frac{\partial f}{\partial y}$	Partial derivative of function FN with respect to y-coordinate
DFNDZ	$\frac{\partial f}{\partial z}$	Partial derivative of function FN with respect to z-coordinate

Mnemonic Variation	Symbol	Comment
DGDY	$\frac{\partial G}{\partial y}$	Partial derivative of function G with respect to y-coordinate
DGDZ	$\frac{\partial G}{\partial z}$	Partial derivative of function G with respect to z-coordinate
DY	dy	Change in y-coordinate along bicharacteristics
DZ	dz	Change in z-coordinate along bicharacteristics
DDY		Change in y-coordinate of bicharacteristics base point as obtained from Newton-Raphson procedure to converge the base point location
DDZ		Change in z-coordinate of bicharacteristic base point as obtained from Newton-Raphson procedure to converge the base point location
FN		Function representing equation of Mach forecone
G		Function representing difference in fitted shock radius and bicharacteristic base point radius for Newton-Raphson procedure
IT = IP		New shock point storage location
QAVE		Average velocity along bicharacteristic
SSAVE		Average sound speed along bicharacteristic
YS(N)		y-coordinate of shock point N.
ZS(N)		y-coordinate of shock point N
SUBROUTINE DIVIDE		
IIPT		Storage index of every other ring of field points off body

Mnemonic Variation	Symbol	Comment
IPTMIN		Number of storage data rings remaining after every other ring is dropped
SUBROUTINE FIELD		
C1AV C2AV C3AV		Average direction numbers for streamline direction
DELXI		Length of a bicharacteristic
IT		Storage index for final field point ring
IP = IK		Storage index for initial field point ring
NP = IN		Storage index for field point ray
SUBROUTINE INTPLT		
RFIT OMEG		Scaled polar coordinates (ρ, ω) of bicharacteristic base point to be used in evaluating interpolating polynomial
SUBROUTINE SECTB		
BT		Function representing difference between actual body radius and body radius computed from present approximation to body point
CP1 CP2 CP3		Direction numbers of normal to normal plane to body containing streamline direction from initial body point
DELY		Change in y-coordinate of new body point
DELZ		Change in z-coordinate of new body point

Mnemonic Variation	Symbol	Comment
FN		Function representing normal plane to body containing velocity vector
IK		Storage index for initial body ring
IP = IT		Storage index for final body ring (IP = 1)
NP		Storage index for ray on which body point is located
SUBROUTINE SETUP		
AM(J, 1)		Coefficient of pressure in the compatibility equation for bicharacteristics (J)
AM(J, 2)		Coefficient of flow angle, θ, in the compatibility equation for bicharacteristic (J)
AM(J, 3)		Coefficient of flow angle, ψ, in the compatibility equation for bicharacteristic (J)
DX, DY, DZ		Distance matrices for coordinate directions along bicharacteristics
AM(J, 4)		Coefficient for constant term in the compatibility equation for bicharacteristic (J)
DELAV		Average parametric angle σ corresponding to bicharacteristic curves.
SDELAV		Sine of DELAV
C4		
C5		
C6		
C7		Average coefficients for compatibility equations
C8		
C9		
C10		
D1-D9		Minors required to solve for $\frac{\partial \theta}{\partial x}$, etc. in EQN 86
S1BAR, S2BAR		Temporary storage

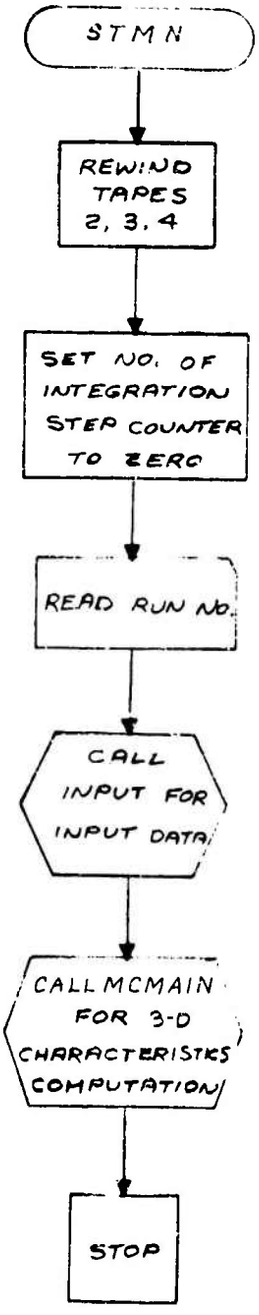
Mnemonic Variation	Symbol	Comment
SUBROUTINE SHOCK		
ALFT2		Present values of direction cosines of normal to shock with respect to y and z axis respectively
ALFT3		
ALFT21		Values of ALFT2 and ALFT3 from previous iteration
ALFT31		
ALFT22		Values of ALFT21 and ALFT31 from previous iteration
ALFT32		
ERRT		Temporary storage for ERROR as interpreted in COMMN/LOOP
ERROR		Difference between average pressure computed from compatibility equations and the pressure computed from shock relations
ERROR1		Previous value of ERROR from last iteration
IP = IK		Storage index for initial shock ring
IT		Storage index for final shock point ring
NP		Storage index for ray on which shock point is located
SUBROUTINE STREAM		
DTS		Change in temperature along streamline
IK		Storage index for initial point data ring
IT		Storage index for final point ring
IN		Storage index for ray on which field point or body point is located
Q2O2		Velocity squared divided by 2
ROINAV		Inverse of average density along streamline
TLAST		Initial value of temperature along streamline

Mnemonic Variation	Symbol	Comment
SUBROUTINE STOP		
NSTOP		Integer test constant for stopping integration if X > XMAX or NPLANE > NPLMAX
NFILE		Number of data planes stored on binary tape 2
SUBROUTINE FLOW		
CPAV		Frozen specific heat behind shock
DHDP		Change in total enthalpy divided by change in pressure
DUN		Change in velocity component normal to shock, across shock
ERR		Difference between free stream, total enthalpy and computed total enthalpy across shock
HT		Total enthalpy behind shock
HPREV		Previous value of HT for previous iteration
PREVP		Previous value of pressure behind shock from previous iteration
QS COMMON AFT3D		Velocity behind shock
TIP		Temperature behind shock
UNAM		Free stream velocity component normal to shock
UNIP		Component of velocity behind shock in direction normal to shock
VTIP		Component of velocity behind shock in direction tangent to shock
USP VSP WSP		x, y and z velocity components upstream of shock

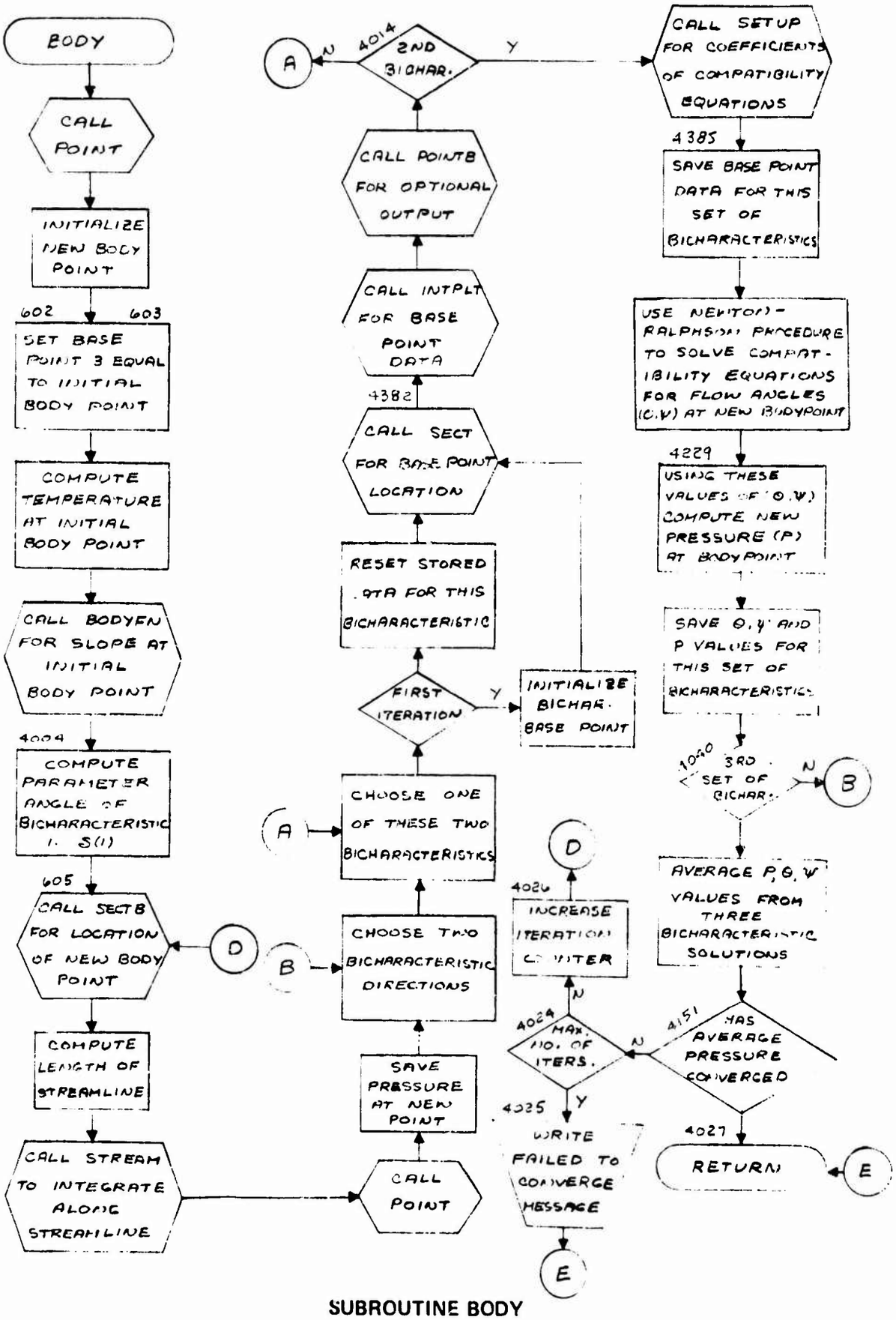
Mnemonic Variation	Symbol	Comment
US VS WS	COMMON AFT3D	x, y and z velocity components across shock

A.7 LOGICAL FLOW DIAGRAMS

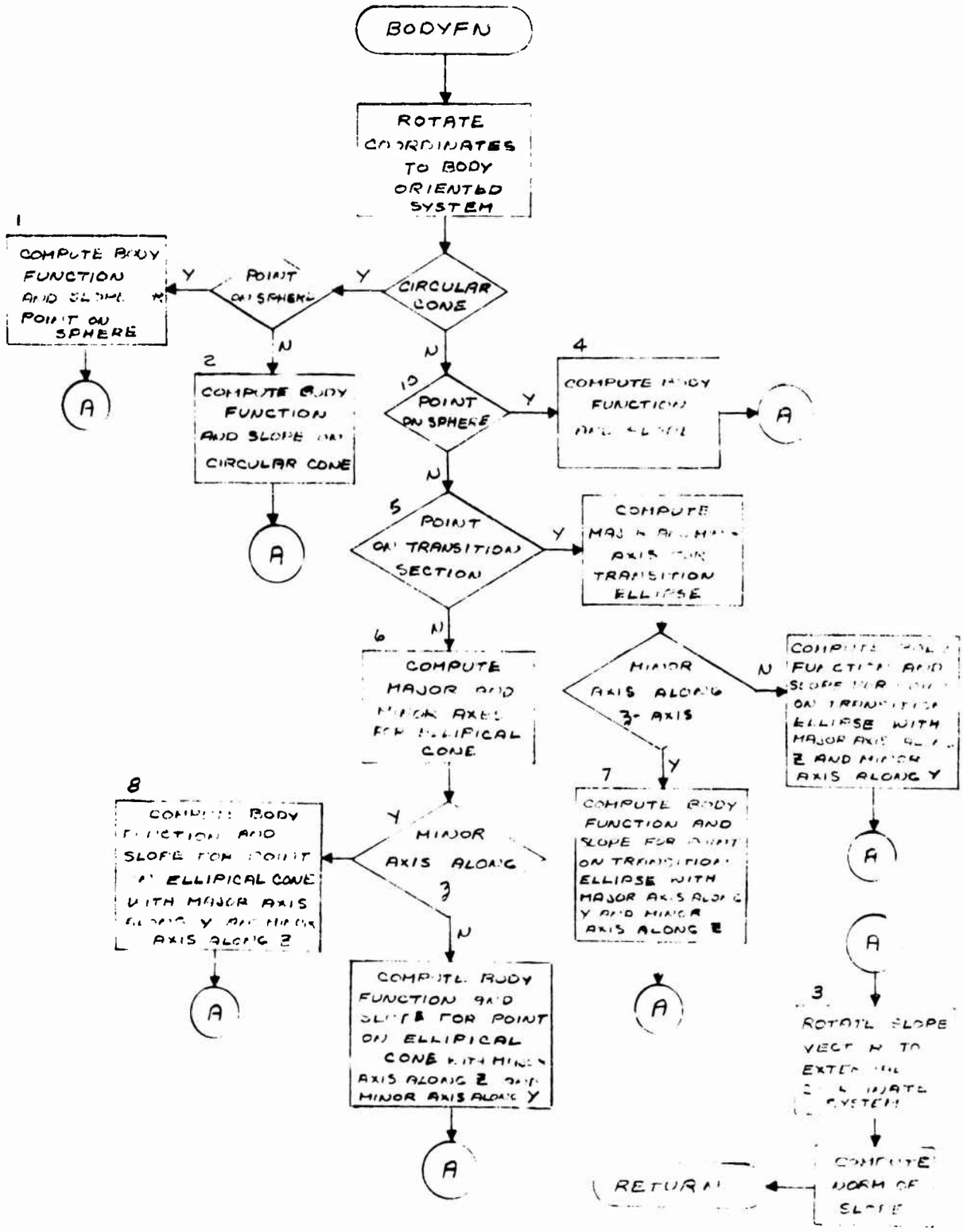
Detailed flow diagrams including all the equations and symbols are probably not so valuable as logical charts when the program is written in FORTRAN IV as is the case here. For this reason only the general logical flow diagrams are presented with no attempt being made to be complete in every detail.



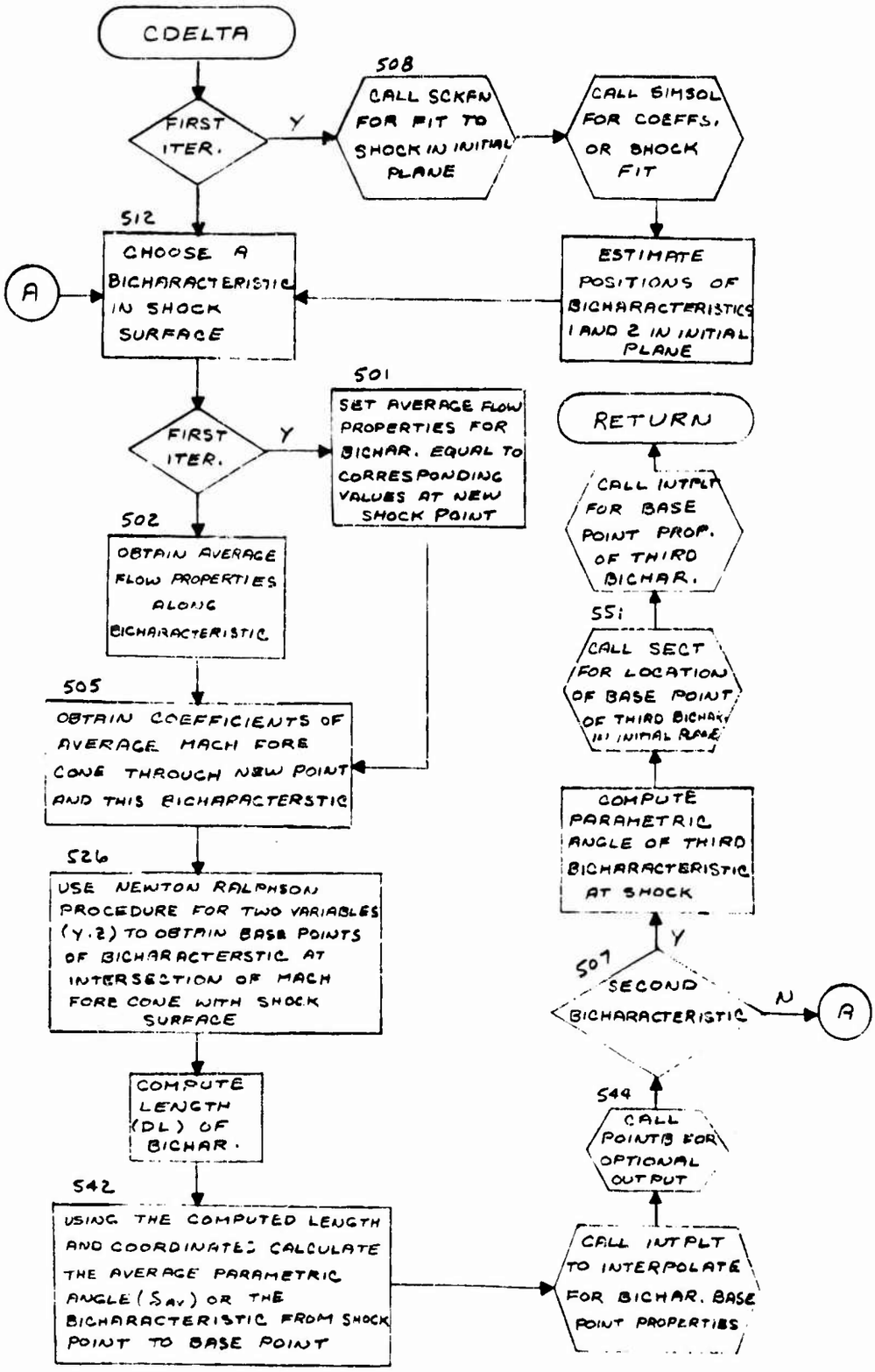
MAIN PROGRAM



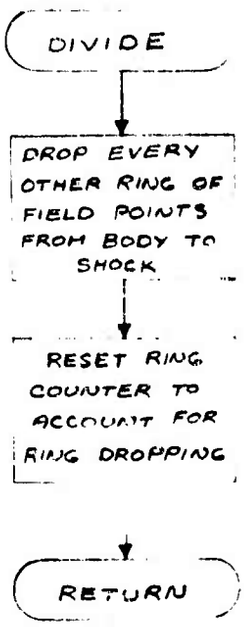
SUBROUTINE BODY



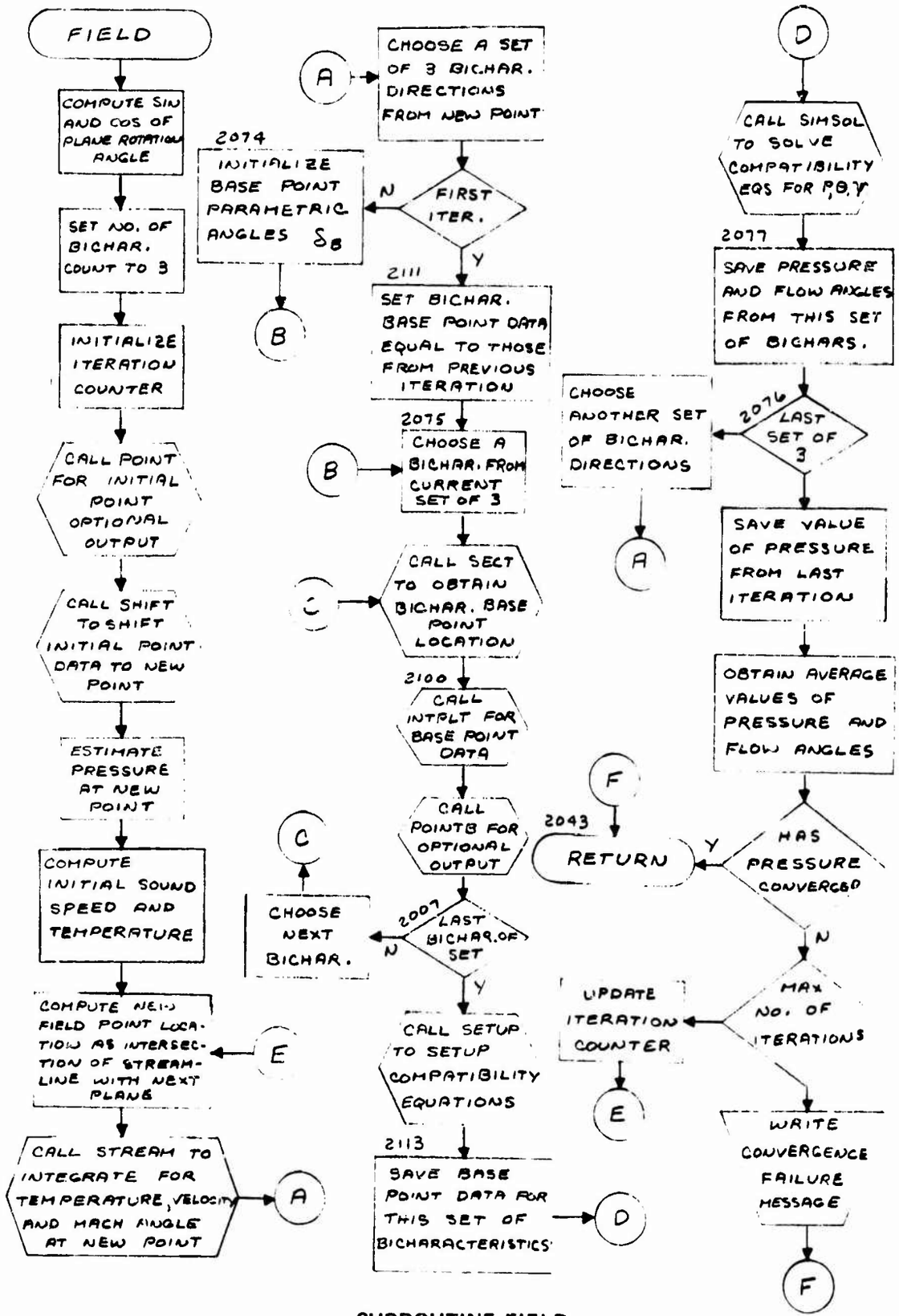
SUBROUTINE BODYFN



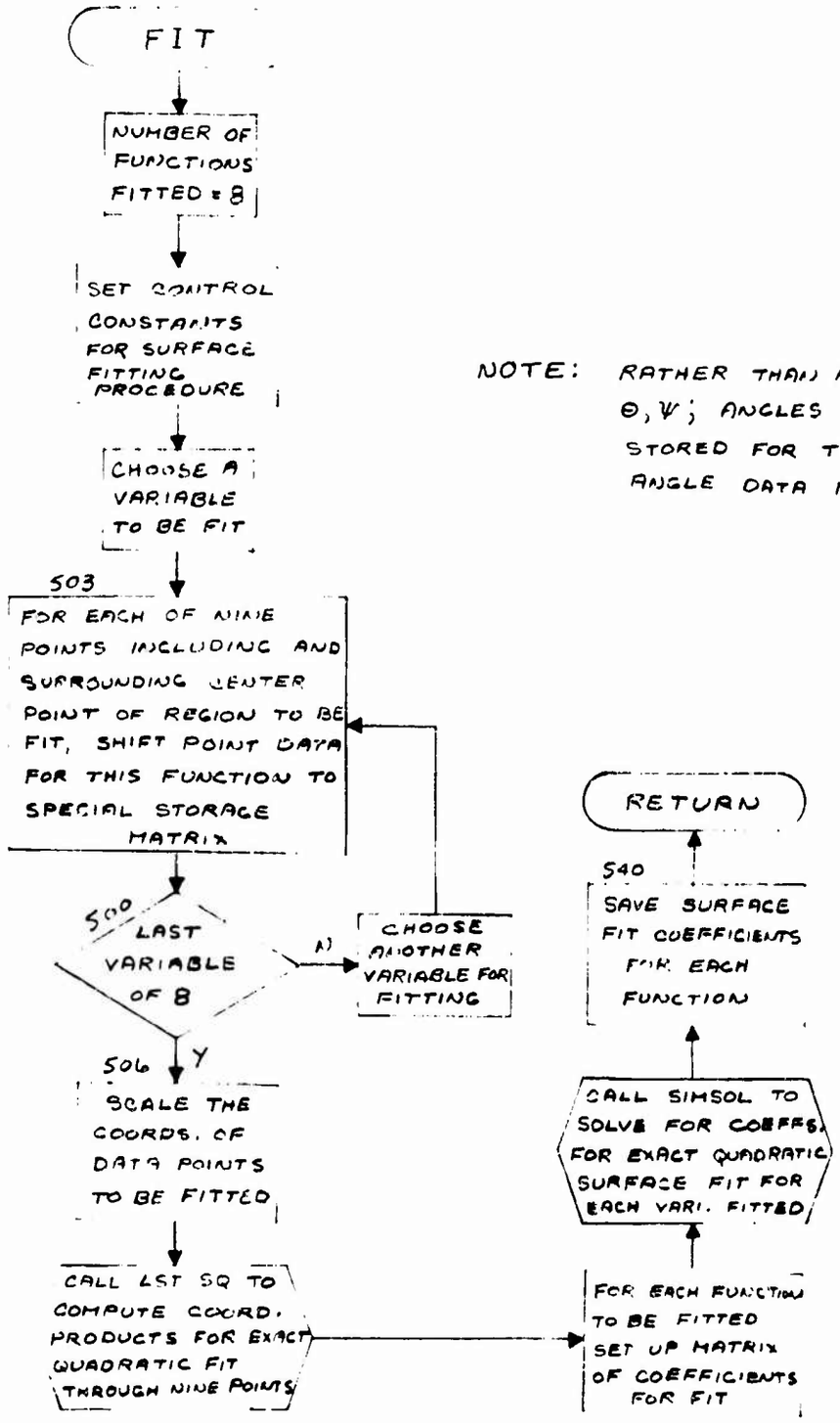
SUBROUTINE CDELTA



SUBROUTINE DIVIDE

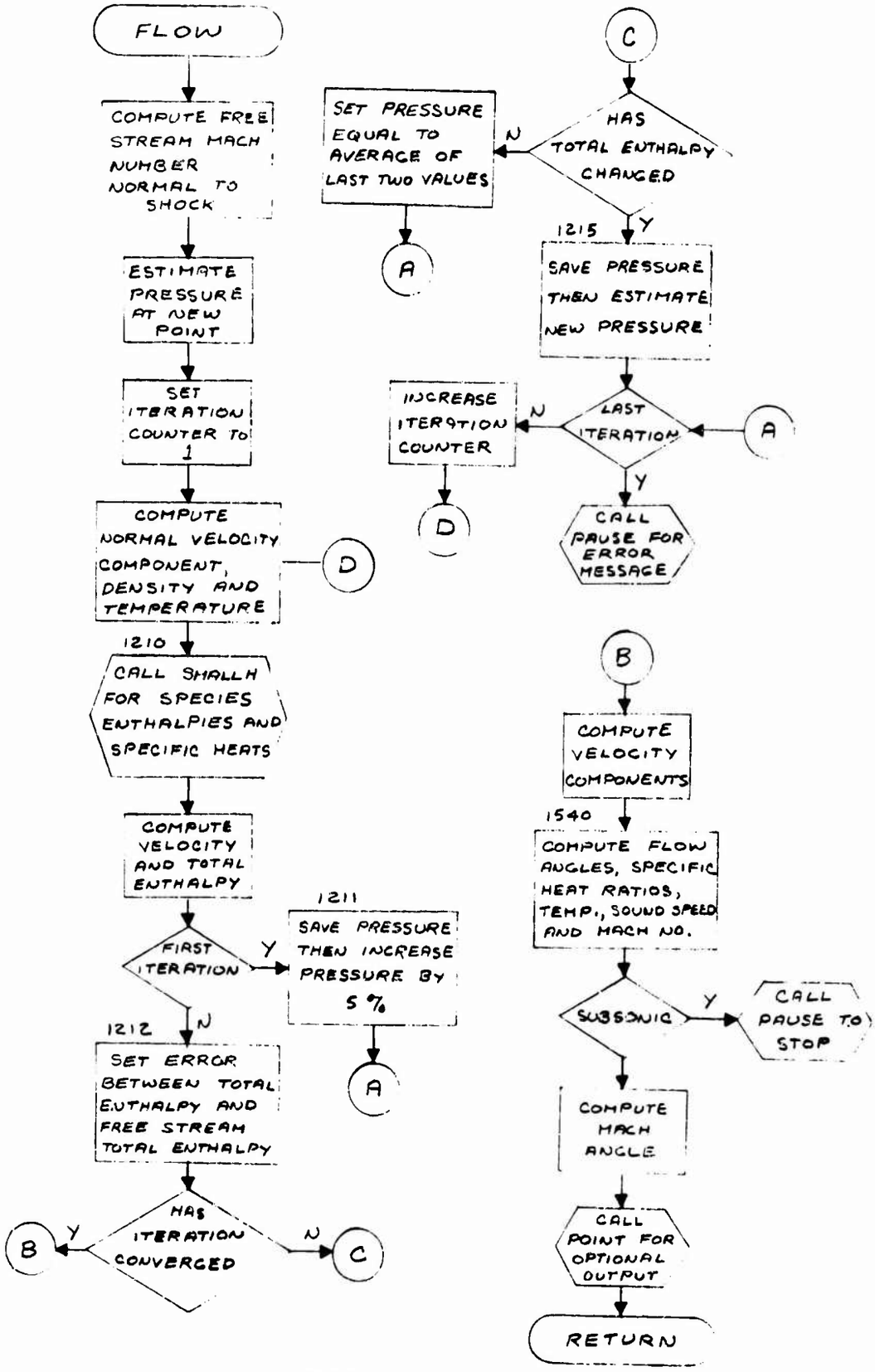


SUBROUTINE FIELD

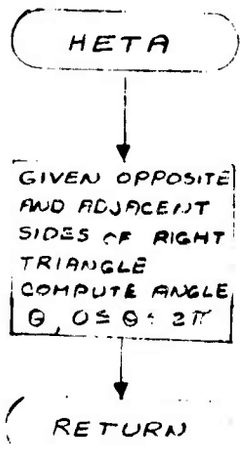


NOTE: RATHER THAN ANGLES Θ, Ψ; ANGLES ω, χ ARE STORED FOR THE FLOW ANGLE DATA MATRIX.

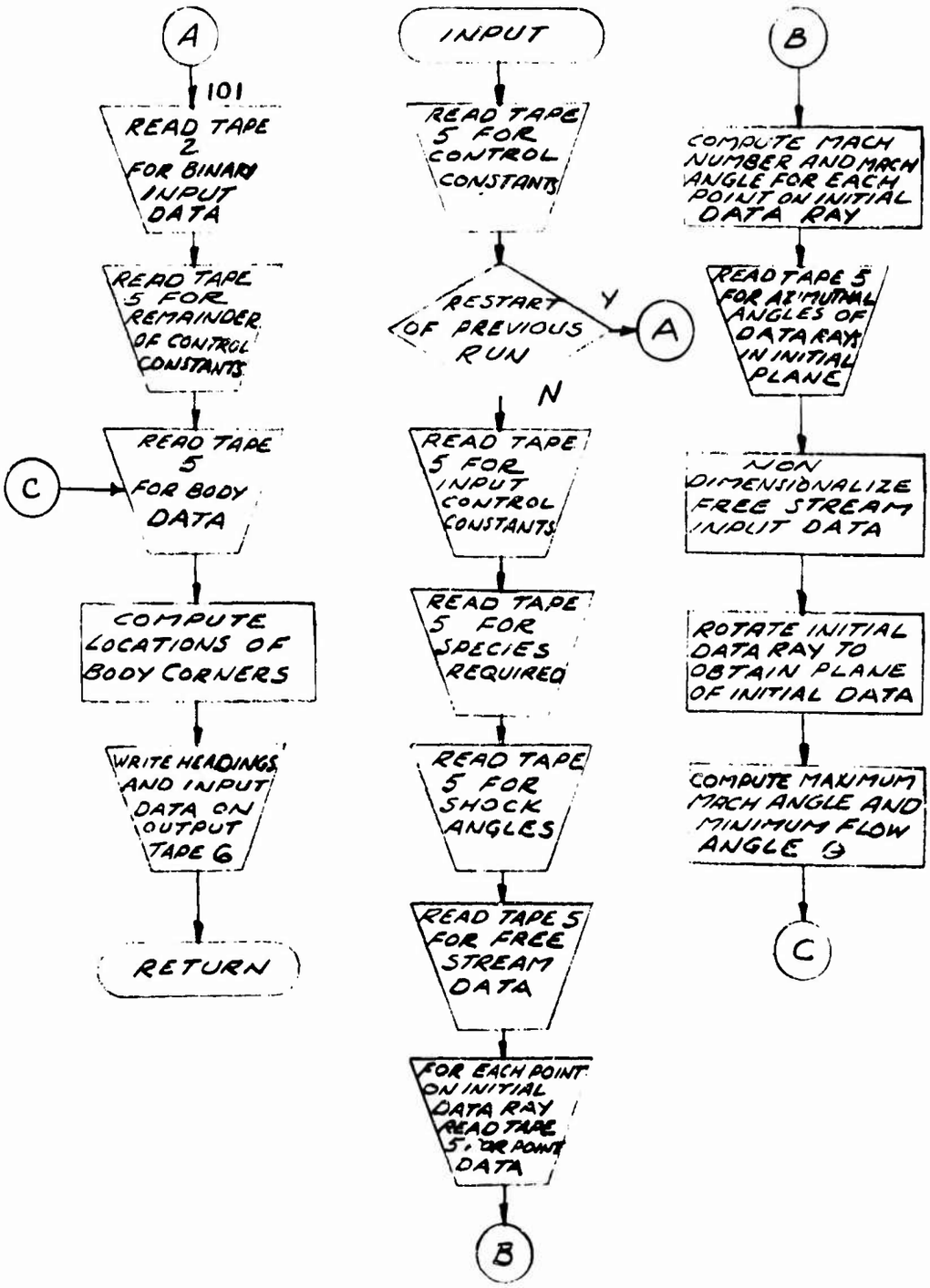
SUBROUTINE FIT



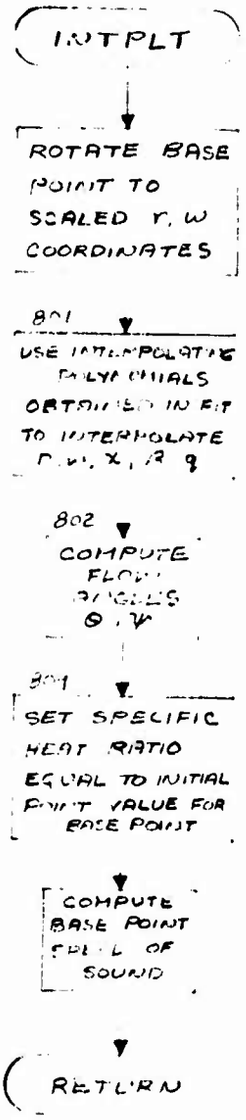
SUBROUTINE FLOW



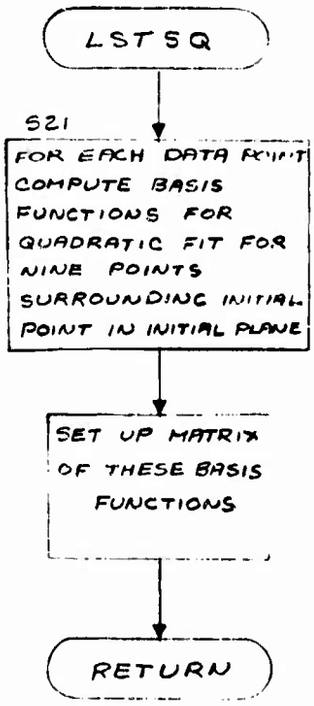
SUBROUTINE HETA



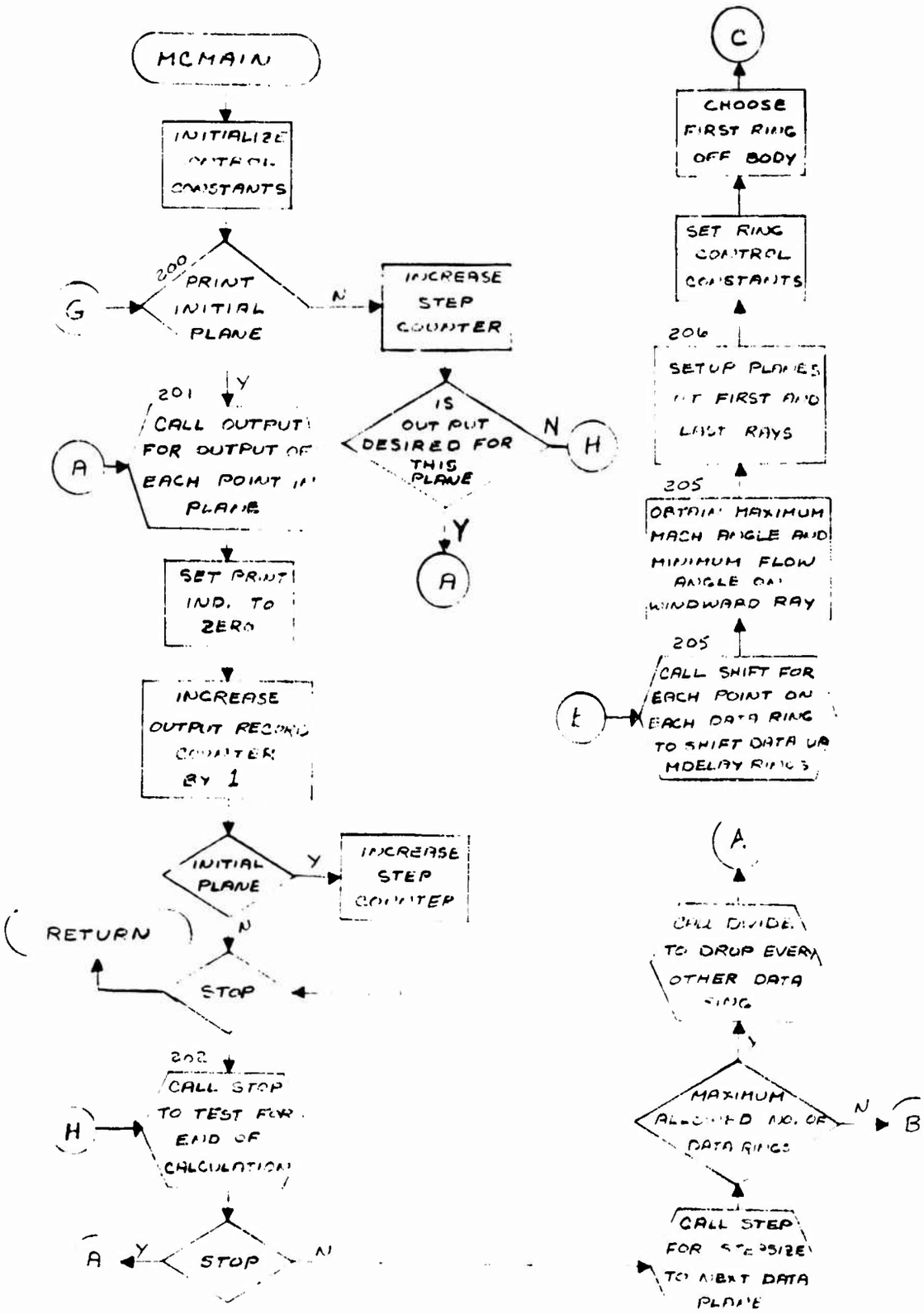
SUBROUTINE INPUT



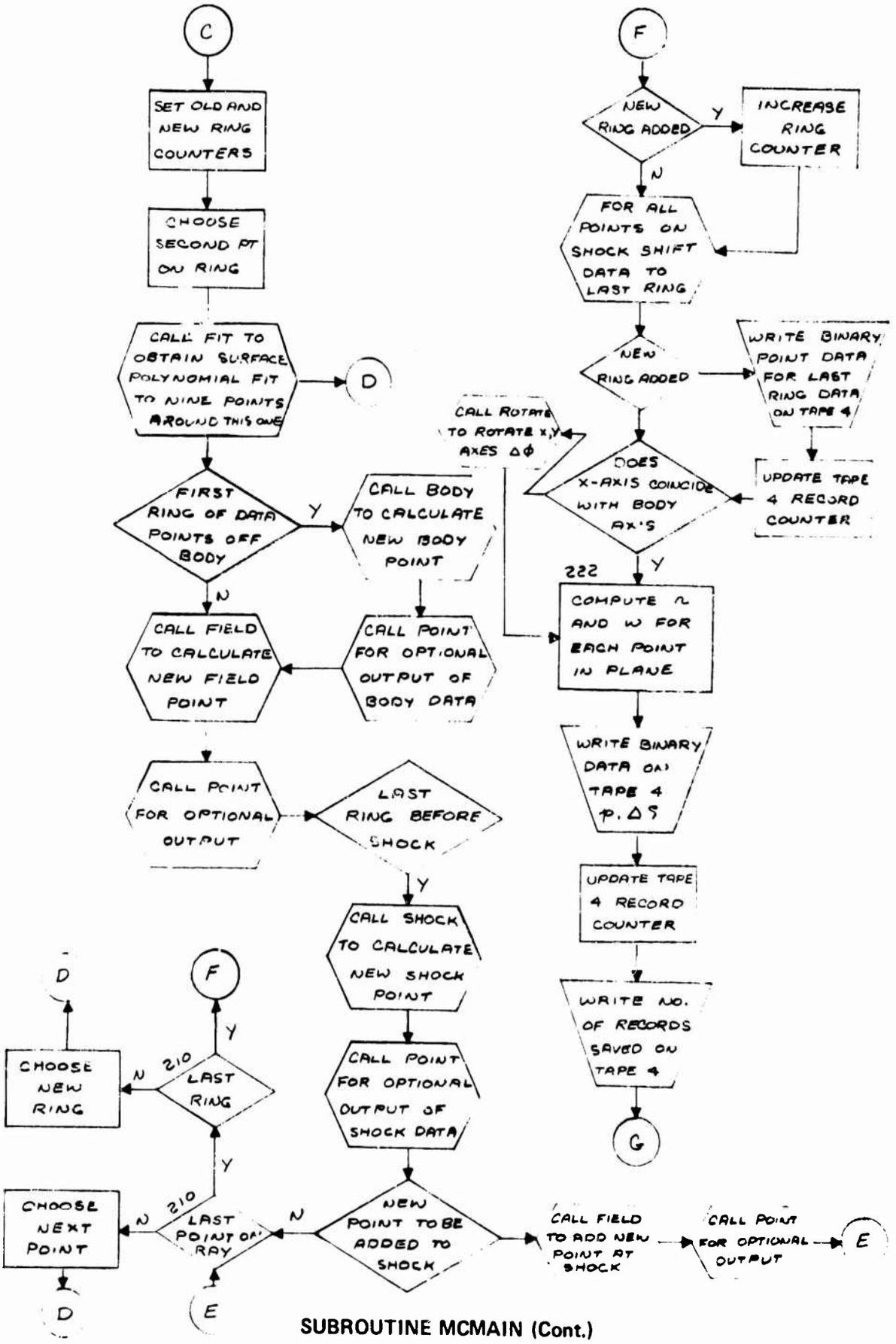
SUBROUTINE INTPLT



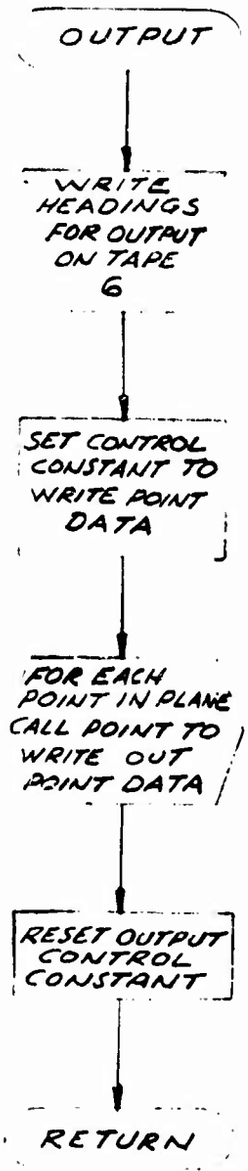
SUBROUTINE LSTSQ



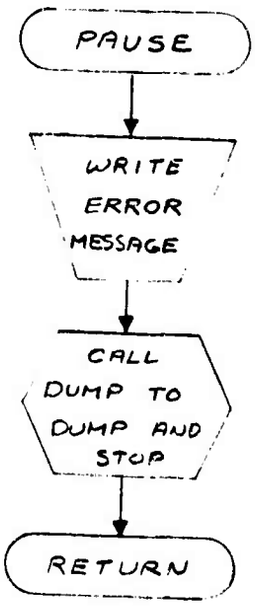
SUBROUTINE MCMAIN



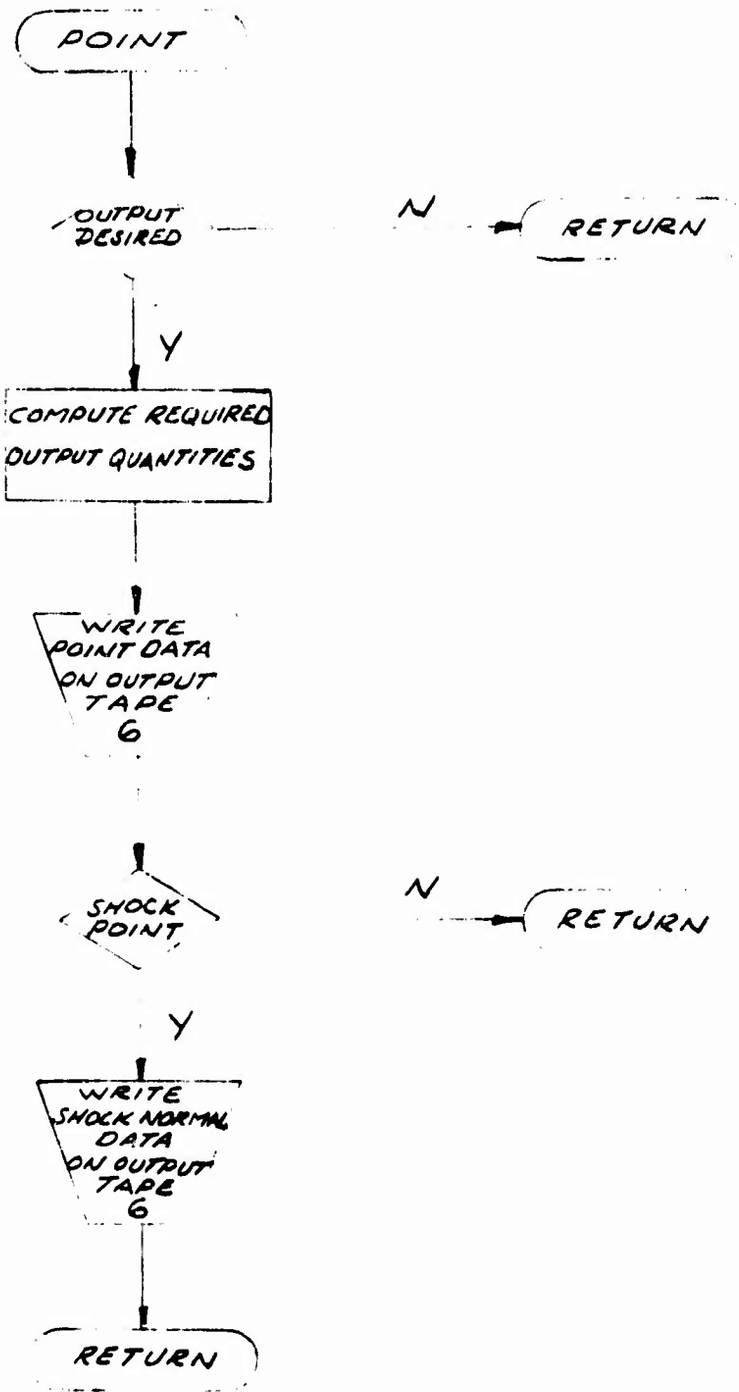
SUBROUTINE MCMAIN (Cont.)



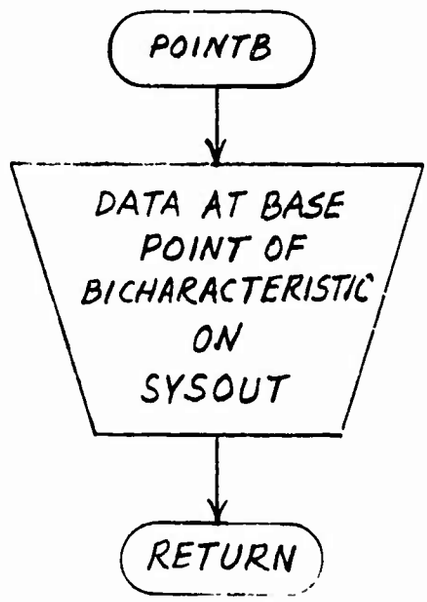
SUBROUTINE OUTPUT



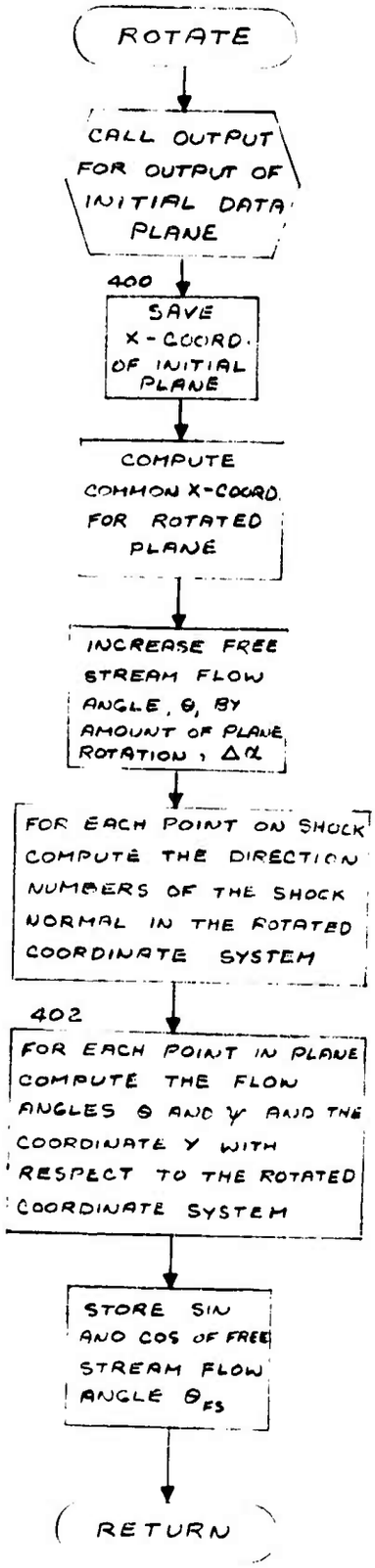
SUBROUTINE PAUSE



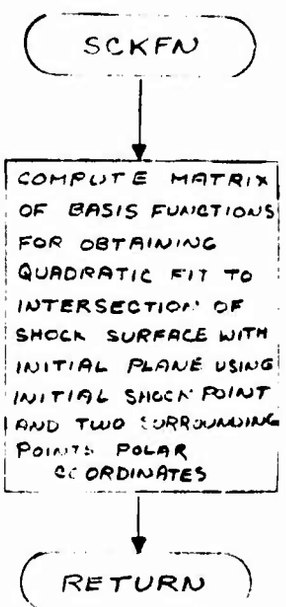
SUBROUTINE POINT



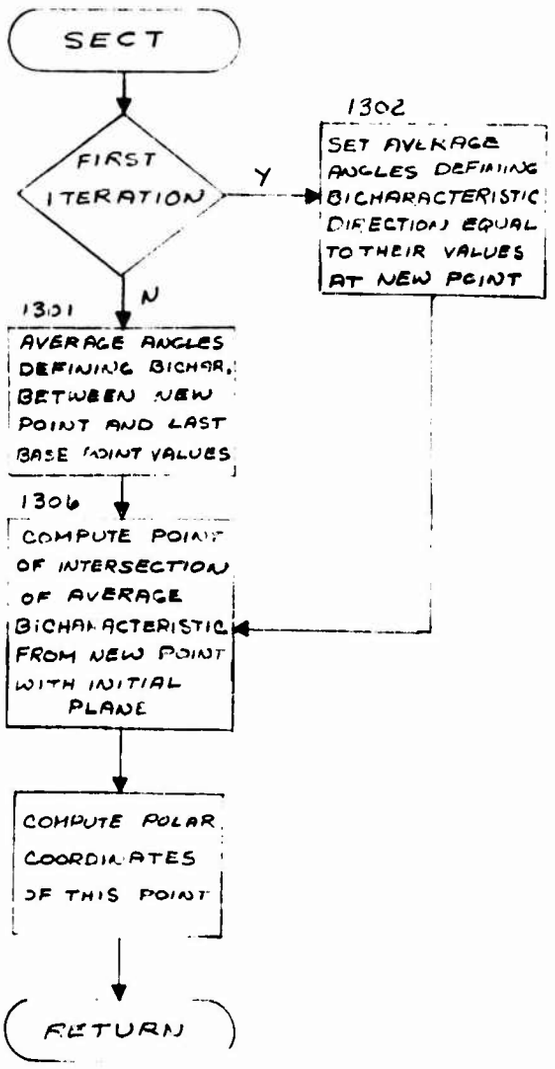
SUBROUTINE POINT B



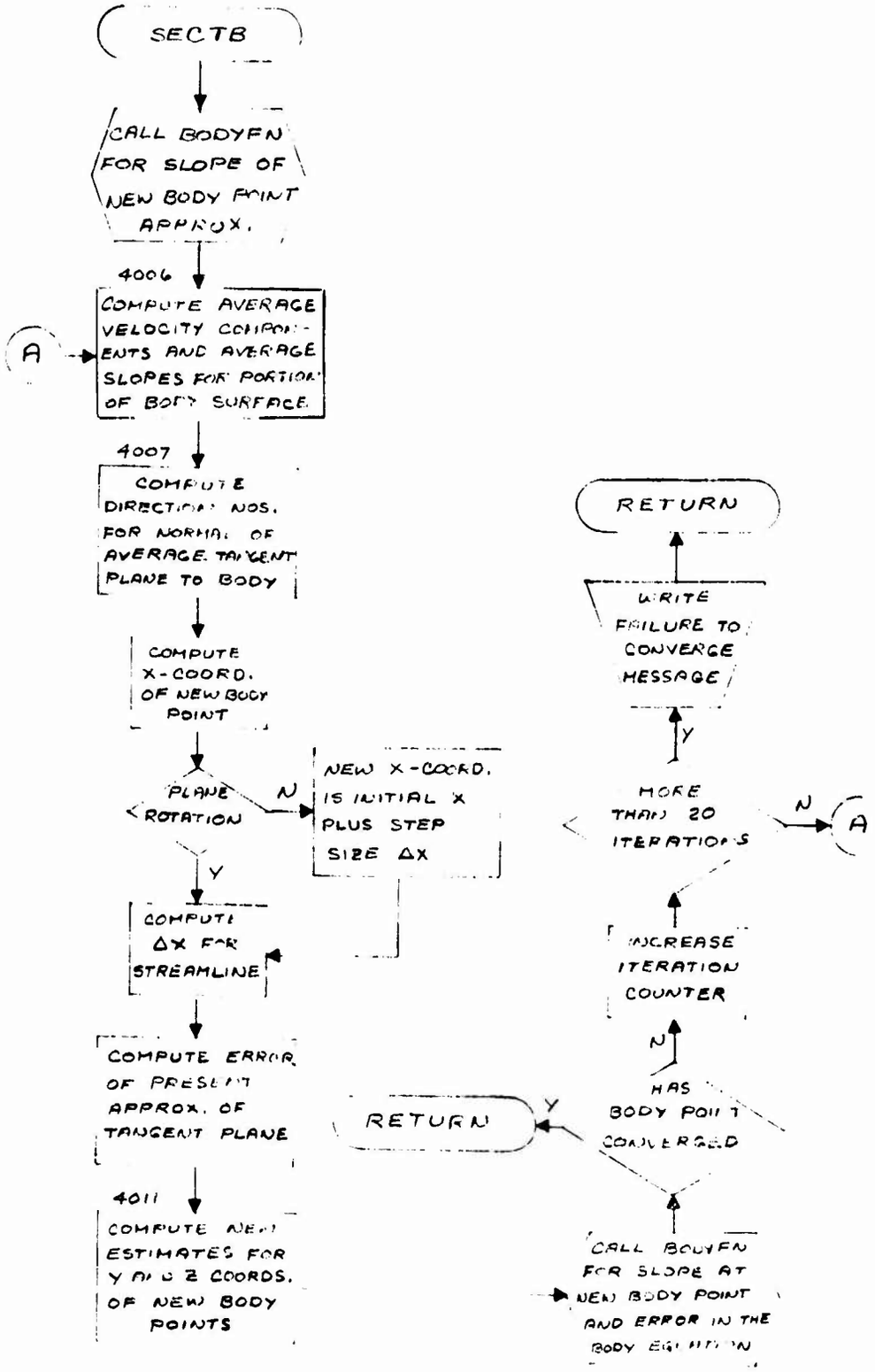
SUBROUTINE ROTATE



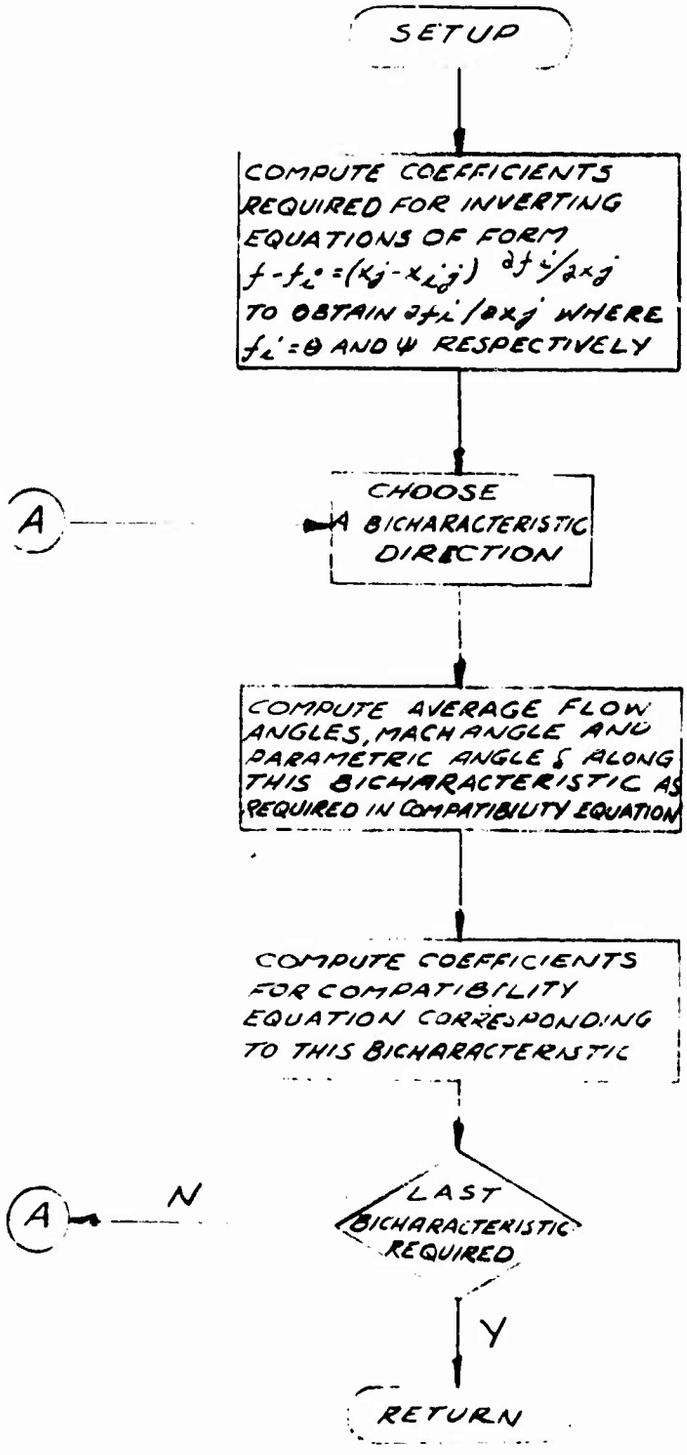
SUBROUTINE SCKFN



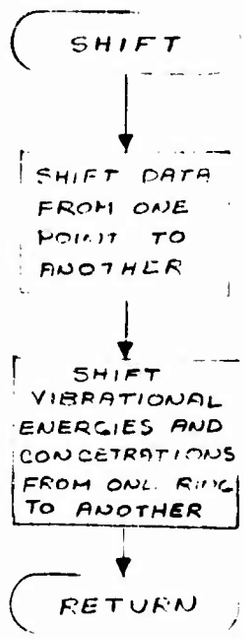
SUBROUTINE SECT



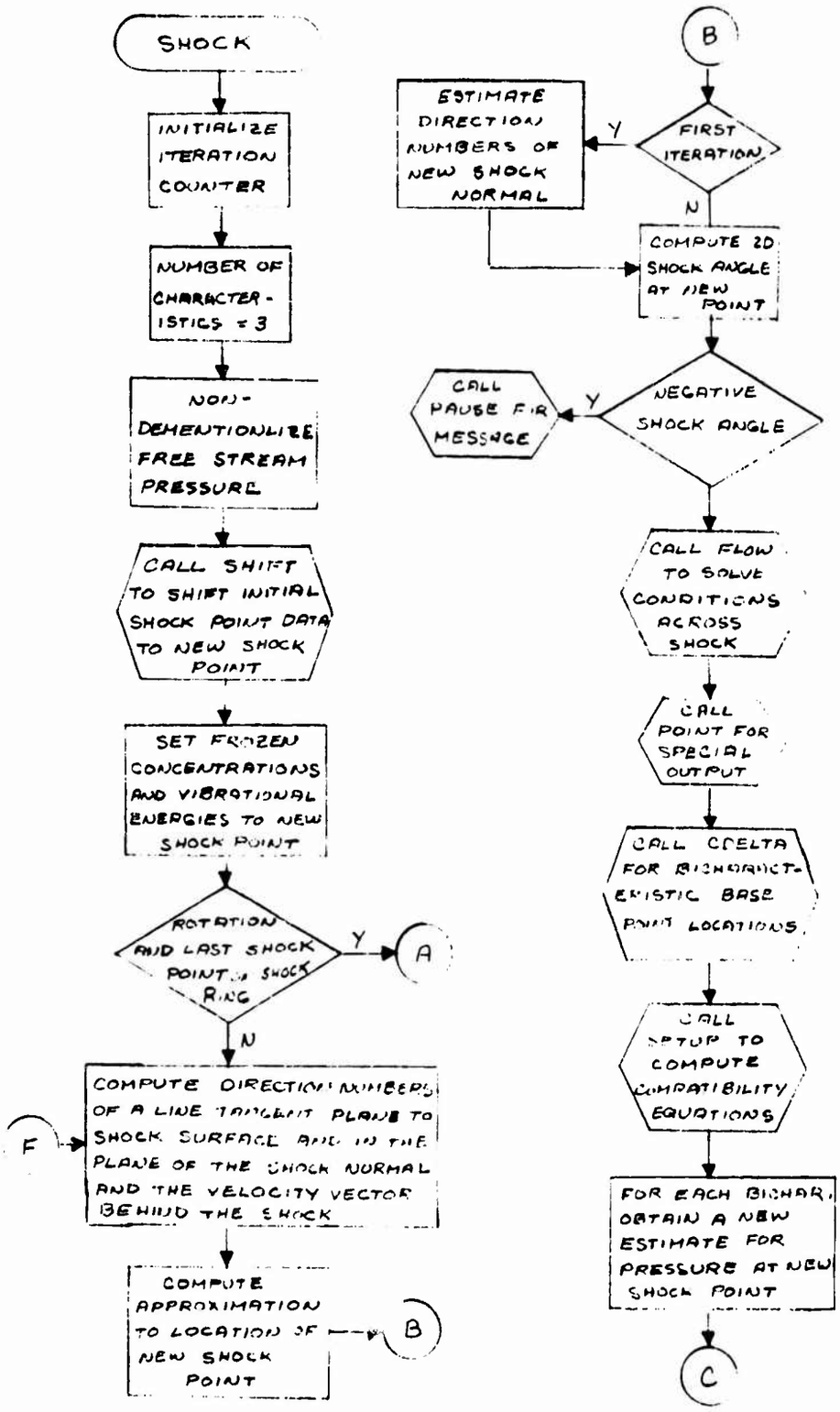
SUBROUTINE SECTB



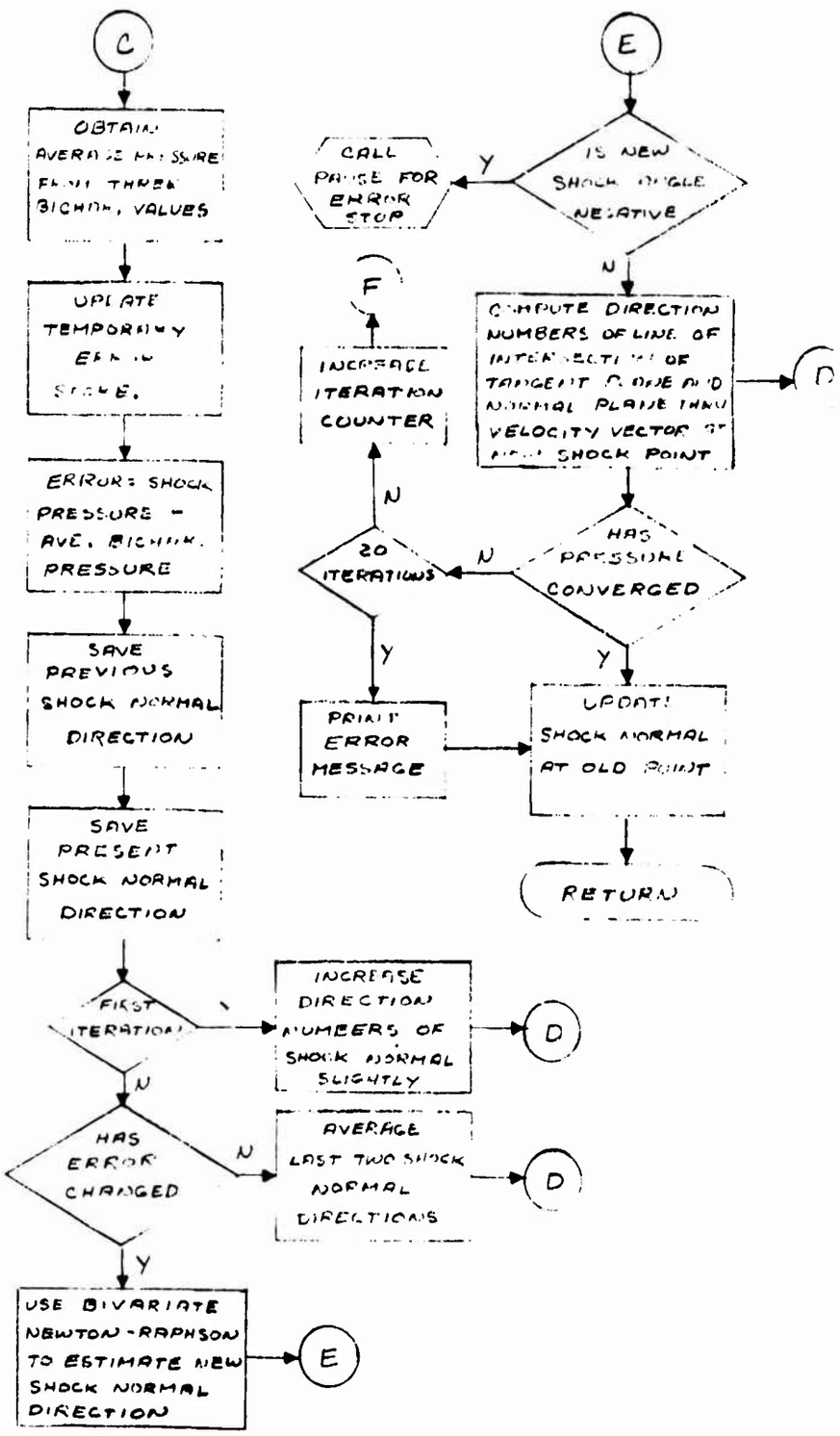
SUBROUTINE SETUP



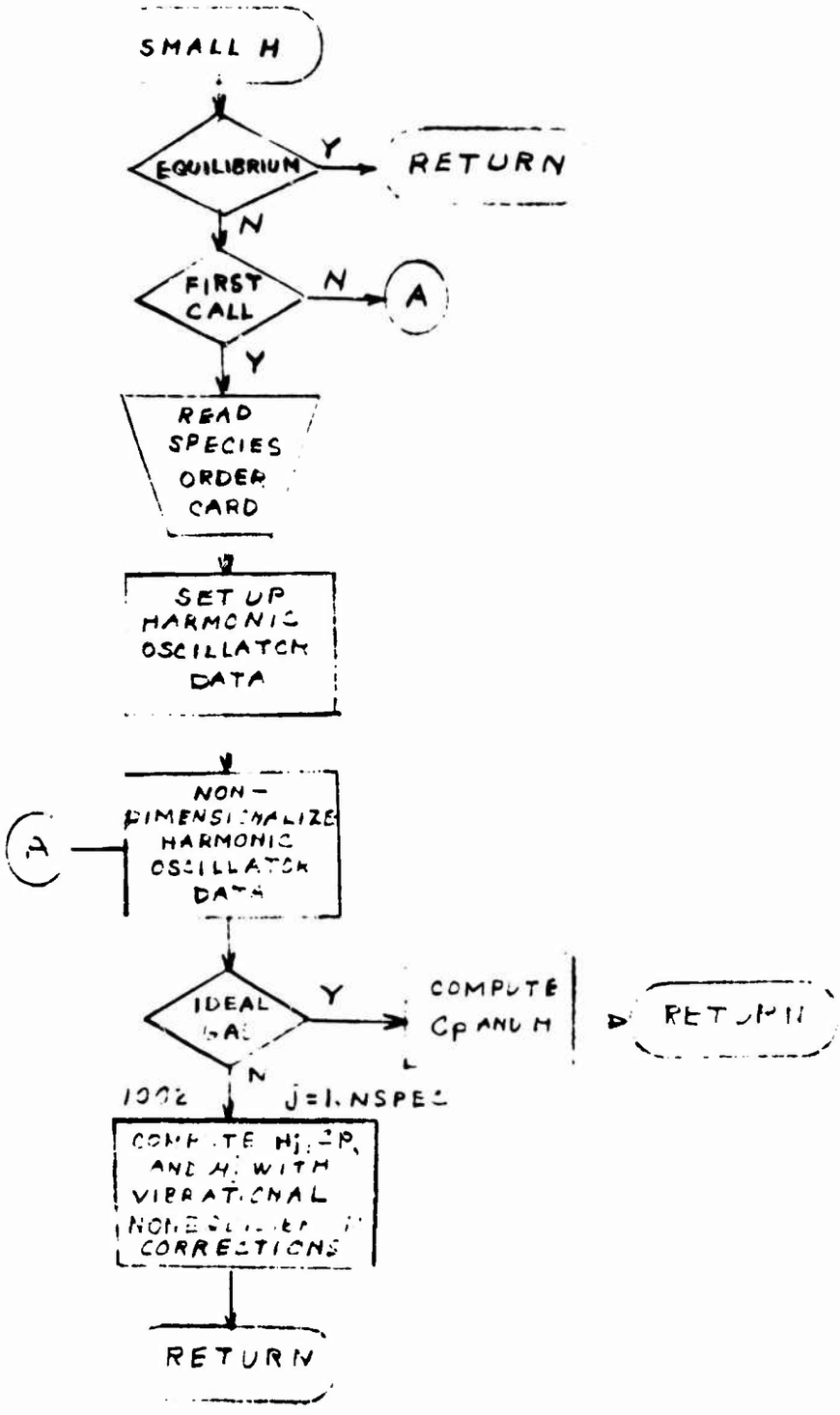
SUBROUTINE SHIFT



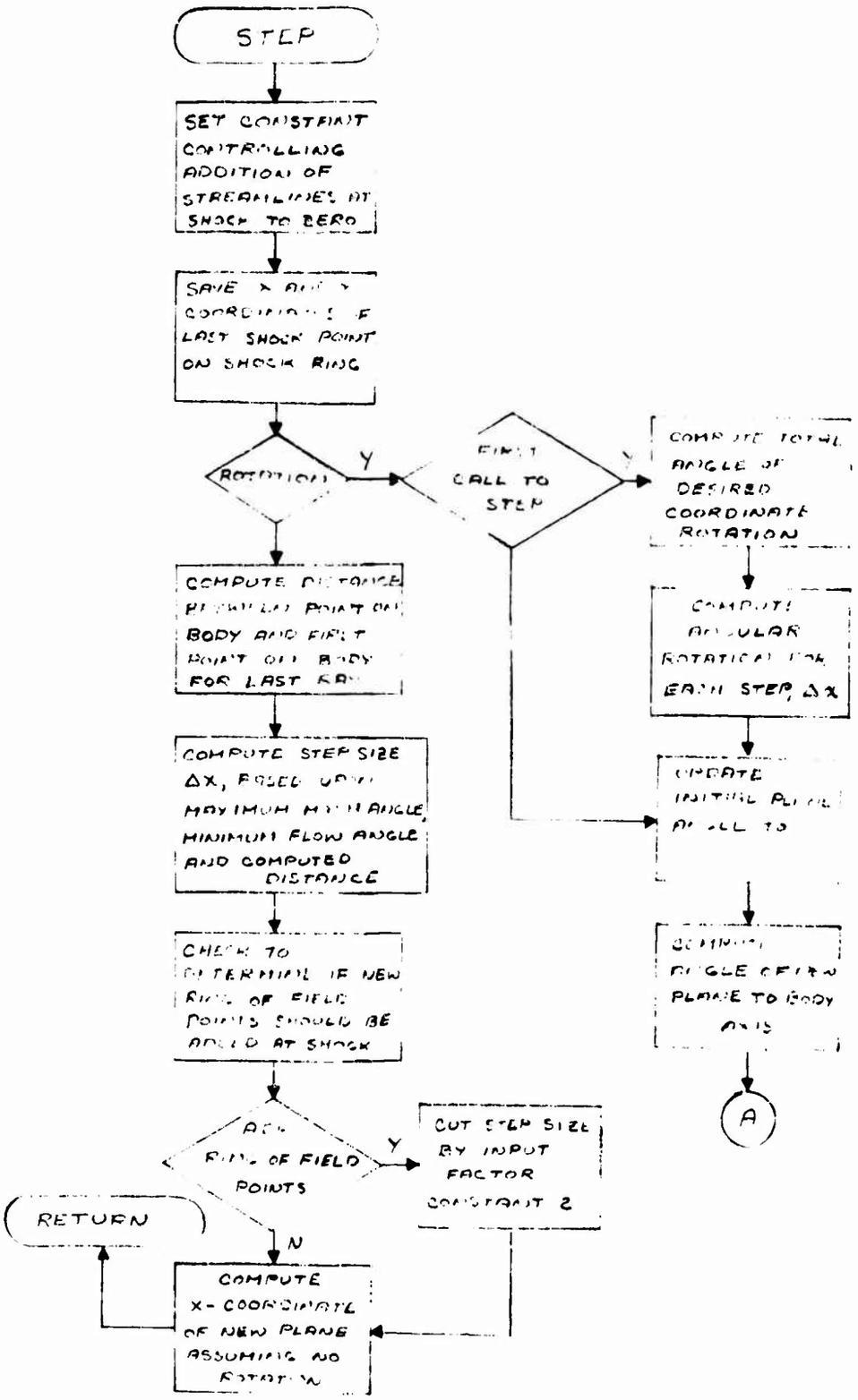
SUBROUTINE SHOCK



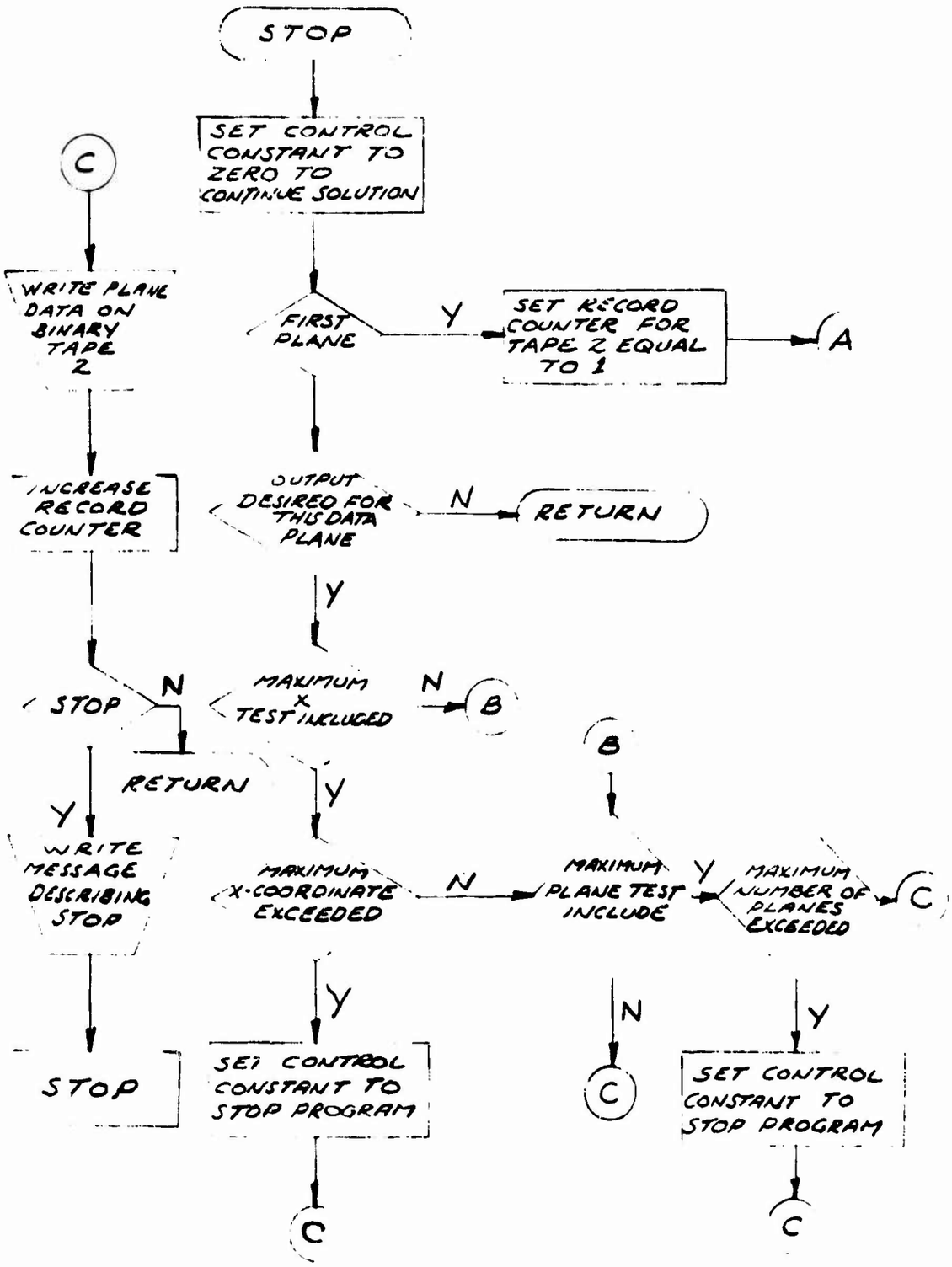
SUBROUTINE SHOCK (Cont.)



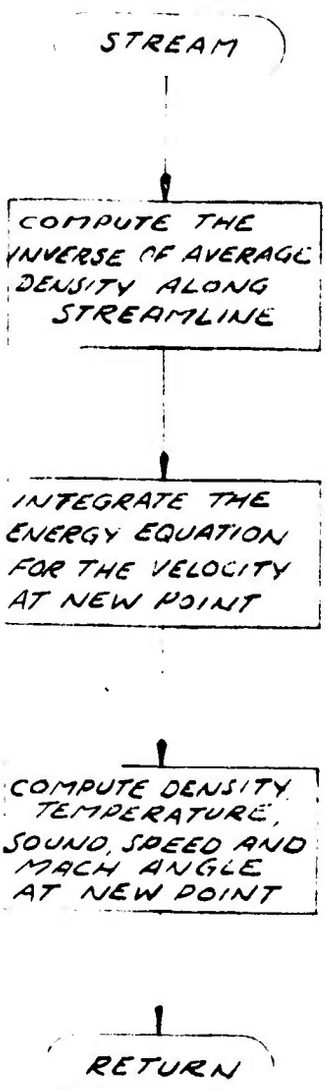
SUBROUTINE SMALLH



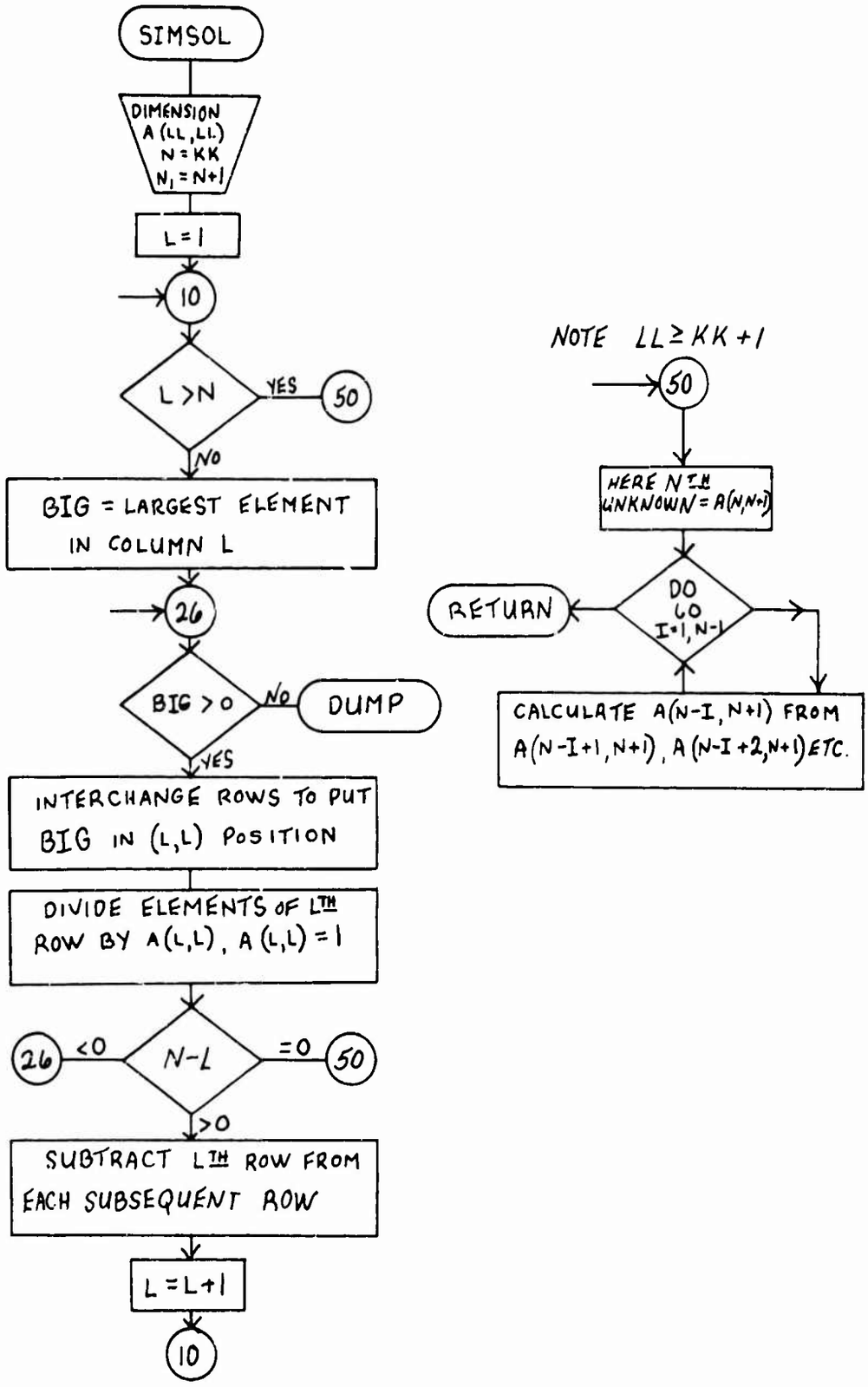
SUBROUTINE STEP



SUBROUTINE STOP



SUBROUTINE STREAM



SUBROUTINE SIMSOL (AK KK, LL)

Appendix B

CALCULATIONS OF FIELD, BODY, AND SHOCK POINTS

TABLE B-I

CALCULATIONS AT A FIELD POINT

Step Size Δx = 0.01460736

Initial Point Data: (All Quantities Nondimensionalized)

x	y	z
-0.45340951	0.75886150	0.55134514

p	q	θ (Radians)	ψ (Radians)
0.21346586	0.62946524	0.3839265	0.2978896

Final Point

	First Iteration	Second Iteration	Third Iteration	Fourth Iteration
x:	-0.43880215	-0.43880215	-0.43880215	-0.43880215
y:	0.76503433	0.76493999	0.76493883	0.76493885
z:	0.55582996	0.55576114	0.55576062	0.55576064
p:	0.20518689	0.20505461	0.20505998	0.205059835
q:	0.63728678	0.63545944	0.63556746	0.635562873
θ:	0.37418336	0.37405871	0.37406140	0.37406141
ψ:	0.28928342	0.28921835	0.28922113	0.28922114

Convergence Test Criterion

$$\frac{p^{(n)} - p^{(n-1)}}{p^{(n)}} \leq 0.00001$$

TABLE B-II

CALCULATIONS AT A BODY POINT

Step Size Δx = 0.01460736

Initial Point Data:

x	y	z
-0.45340951	-0.44568498	0.77194911

p	q	θ	ψ
0.19598220	0.61962985	-0.22850411	0.41459165

Final Body Point:

	First Iteration	Second Iteration	Third Iteration
x	-0.043880215	-0.043880215	-0.043880215
y	-0.44928833	-0.44929017	-0.4492910
z	0.77819028	0.77819325	0.77819480
p	0.18682411	0.18683943	0.18683876
q	0.62771747	0.62712205	0.627113444
θ	-0.22122613	-0.22119960	-0.22119938
ψ	0.40008304	0.40009536	0.40009460

TABLE B-III

CALCULATIONS AT A SHOCK POINT

Step Size Δx = 0.01460736

Initial Point Data:

x	y	z
-0.45340951	1.2645526	0.0

p	q	θ	ψ
0.34633269	0.77348271	0.53874810	0.0

N_x	N_y	N_z
-0.64625884	0.76311828	0.0

where $\bar{N}$ = normal vector to shock

Final Point:

	First Iteration	Second Iteration	Third Iteration
x	-0.43880215	-0.43880215	-0.43880215
y	1.2769231	1.2764755	1.2768739
z	0.0	0.0	0.0
p_c	0.16793336	0.17218783	0.15803909
p_s	0.14470096	0.13455693	0.16085586
q	0.21948106	0.22696227	0.20741885
θ	0.52119923	0.51189300	0.53575372
ψ	0.0	0.0	0.0
N_x	-0.61823331	-0.64325045	-0.64156076
N_y	0.78599464	0.76565583	0.76707221
N_z	0.0	0.0	0.0

TABLE B-III (Cont.)

	Forth Iteration	Fifth Iteration
x	-0.43880215	-0.43880215
y	1.2768463	1.2768457
z	0.0	0.0
p_c	0.15898510	0.15900633
p_s	0.15904675	0.15900626
ρ	0.20877892	0.20880932
θ	0.53413956	0.53410338
ψ	0.0	0.0
N_x	-0.64152290	-0.64152295
N_y	0.76710389	0.76710384
N_z		0.0

Convergence Test Criterion

$$\frac{p_c^{(n)} - p_s^{(n)}}{p_s^{(n)}} < 0.00001$$

where

p_c = average pressure from compatibility equation

p_s = pressure from shock relations

Appendix C

AUTOMATED INPUT PREPARATION

For certain configurations, the flow over the nose region may be described by two space variables. In these cases, the preparation of input data for the present three-dimensional program will be facilitated by certain modifications of the axisymmetric program described in Reference 60. For convenience, FORTRAN listings of the relevant portions of the referenced program are reproduced. The shaded statements indicate the changes.

```
*IBFTC STOIA   LIST,DECK                                                AFTB0000
      SUBROUTINE START                                                  AFTB0010
      COMMON/CMBF/ TW, CMALL,EOICT,CDCT                                 AFTB0020
      COMMON/THST/ WCBT(7,15),ECBT(7,15),PSCBT(7,7),WJO(15),            AFTB0030
     1             H(15),F(15),CP(15),BPSES(2),WBL(15)                  AFTB0040
      COMMON / CONTRL/ EPSC,TVIS,IT,NC,ITAPE,NXDIV,DELX,IMOD,JOIN,      AFTB0050
     1             OMEGA,PRCORR,DELT ,TIET,EPSLON,TEST ,  DEL      ,    AFTB0060
     2             TEMP,PRES,DENS,VEL,WGHT,XLAMDA,XLM,.1 , TTEST  ,    AFTB0070
     4             NRUN,NBSIT,IPRNT,NOUT,NPSIST,CDST,ZETAMX,ZMAX,       AFTB0080
     5            INSLR,INDWT,INAFT,NBDY,ZOUTP(48),NRAD,IONII,ISW6,     AFTB0090
     6            SKAP,SLAM,GAMTST,NPTS,NMAX,DRVMAX ,XDELAY, TMALL,     AFTB0100
     7             ITSOM,RMS,DEV,RSHP,SIP,XLAM,XKAP,XCST,XZETAM,        AFTB0110
     8             ERRNEW,TSTGAM,NMMAX,MISW6,ZMAXP,NOUTN  ,             AFTB0120
     9            MINVAR , MAXVAR ,  PSIED(31)                         
      COMMON/FREE/ XC,PINF,GAMINF(15),FRSINF(2),VAM,RAM,RHOAM,UAM,      AFTB0140
     1             GAMAM(15),ERSAM(2),TAM,SIGAM,HIMF      , WTAM       
      COMMON/BODY/  ZBDY(6),YBDY(6),SDI(6),SD2(6),SDC(6,6),             AFTB0160
     1             SDIN(6),SD2N(6),SDCN(6,6),ZBDYN(6),YBDYN(6)          
      COMMON/REACT/ THETA(2),TAUA(2),TAUB(2),TAUC(2),TAUD(2),C(15),     AFTB0170
     1             NUP(35,15),NU(35,15),AK(35),BK(35),CK(35),DK(35),    AFTB0180
     2             KFIIND(35),CZI(35),CWI(35),CDI(35),A(35,2),          AFTB0190
     3             ALPIJ(8,15)                                          AFTB0200
      DIMENSION IBREC(1),Y0(1)                                          AFTB0210
      COMMON/AFTBDY/Z(35),R(35),RHO(35),RHOI(35),XMA(35),PSI(35),       AFTB0220
     1   PHI(35),BETA(35),CA(35),CB(35),P(35),T(35),U(35),GAMMA(35,15), AFTB0230
     2   EPS(35,2),CPP(35),CH(35),CHMOL(35),RC(35),B(35),PL(35),QL(35), AFTB0240
     3   DSTL(35),PL(35),ELT(35)                                        AFTB0250
      COMMON/AFTBDY/BZ(3),BR(3),BPHI(3),BDELTA(3),BPSI(3),BPI(3),       AFTB0260
     1   BXI(3),BOMEGA(3),C1S(3),C2S(3),C3S(3),SIGMA(3),H0,IB,IS,I ,    AFTB0270
     2   IST,IS2,IS3,IFI,IF2,IF3,NRAY,CHI,ST,XMD,XCP,ZRAY,ZETA(2),DPDS, AFTB0280
     3   IONSW(8,9),QTDS,SBAR,GOUT,XPI,MAXRAY,IS41,IS44,MCASUT,          AFTB0290
     4   NMOD6,NXE 6,N 004,NXE 4, SOUTY,NTAPE,IBPCH,IBODY, AFIT,PMSUT,  AFTB0300
     5   RUETAT,RHOFL(3,3),DCON(3),PCON(3),IBB(3),PIBAR(3),PIHAT(3),     AFTB0310
     6   PIERR(3),BETAS(3),PHIS(3),PSIBAR(3),DELBAR(3),PSL(3,3),         AFTB0320
     7   B(15),NROTAT,GAMMA0,DEPSDY(2),DGAMDY(15),DEPSDS(4,2),           AFTB0330
     8   DGAMDS(4,15)                                                   AFTB0340
      DIMENSION W(35,1)                                                 AFTB0350
      COMMON/EQUIL/NGAS,NTEST,NUMX,HEO,GAMO                             AFTB0360
      COMMON/FREEZE/IFROZE                                              AFTB0370
      COMMON/ARCLEN/ARC                                                 AFTB0380
      COMMON/CALCD/NPUNCH                                               AFTB0390
      DIMENSION ENTRO(1)                                                AFTB0400
      EQUIVALENCE (CPP,ENTROP)                                          AFTB0410
      COMMON/PROP2/ WTCP                                                AFTB0420
      DIMENSION CM(1)                                                   AFTB0430
      EQUIVALENCE (CM,CMACH)                                            AFTB0440
```

```
      DIMENSION NPHIL(1)                                                AFTB0460
      EQUIVALENCE(IPS,IL,NBAT),(IDPS,NSPEC)                             AFTB0470
C     READ IN DATA AND PRINT OUT THE CONTROLS. INITIALIZES              AFTB0480
C                                                                       AFTB0490
C  READS DATA FROM CARDS OR TAPE TO START AFTERBODY PROGRAM LINK        AFTB0500
C                                                                       AFTB0510
      REWIND 4                                                          AFTB0520
      NREC2 = 3                                                         AFTB0530
      READ (5,1) ITAPE,MRCORD,NREC2N,MPUNCH                             AFTB0550
    1 FORMAT (16I5)                                                     AFTB0560
      IF(INAFT.EQ.2)GO TO 1100                                          AFTB0570
      READ(4) DZ(3),DR(3),DUM,DPHI(3),SBAR,XPI,GAMLL      ,DUM,DUM,     AFTB0580
     1       DUM,DUM,DDELTA(3),DPSI(3),RSAV,ZSAV,PSI2                   AFTB0590
      READ (4) (YBDY(L),ZBDY(L),BD1(L),BD2(L),(BDC(L,K),                AFTB0600
     1K=1,5),L=1,MBDY)                                                  AFTB0610
      READ(4) NPTS,DUM,DUM,DUM,DUM,DUM                                  AFTB0620
      DO 505 L=1,NPTS                                                   AFTB0630
  505 READ(4) EPS(L,1),EPS(L,2),DUM,P(L),RHO(L),U(L),                   AFTB0640
     1      (GAMMA(L,J),J=1,15),XMW(L),T(L),RAD,R(L),Z(L),PSI(L),       AFTB0650
     2       U(L),PHI(L),SIGMA(3)                                       AFTB0660
      GO TO 507                                                         AFTB0670
 1100 READ(5,1110)DZ(3),DR(3),DPHI(3),SBAR,XPI,GAMLL,DDELTA(3),DPSI(3), AFTB0680
     1RSAV,ZSAV,PSI2,(YBDY(L),ZBDY(L),BD1(L),BD2(L),(BDC(L,K),K=1,5),    AFTB0690
     2L=1,NBDY)                                                         AFTB0700
 1110 FORMAT(5E15.7)                                                    AFTB0710
      READ(5,1115)NPTS                                                  AFTB0720
 1115 FORMAT(I5)                                                        AFTB0730
      DO 1120 L=1,NPTS                                                  AFTB0740
 1120 READ(5,1110)EPS(L,1),EPS(L,2),P(L),RHO(L),U(L),(GAMMA(L,J),J=1,15)AFTB0750
     1,XMW(L),T(L),RAD,R(L),Z(L),PSI(L),PHI(L),SIGMA(3)                 AFTB0760
  507 CONTINUE                                                          AFTB0770
      IF(INBLR .EQ.0) GO TO 9034                                        AFTB0780
      DO 9031 N = 1,NPTS                                                AFTB0790
      IF(RSAV .LE. R(1) ) GO TO 9032                                    AFTB0800
      IF(RSAV.GE.R(N).AND.RSAV.LE.R(N+1))GO TO 9032                     AFTB0810
 9031 CONTINUE                                                          AFTB0820
 9032 CONTINUE                                                          AFTB0830
      NP1 = N                                                           AFTB0840
      NP2 = N+1                                                         AFTB0850
      NP3 = NPTS +1                                                     AFTB0860
      DENOM = R(NP1)-R(NP2)                                             AFTB0870
      NSPEC2 = 30                                                       AFTB0880
      DO 9033 J=4,NSPEC2                                                AFTB0890
      AAA =(R(NP1,J)-R(NP2,J))/ DENOM                                   AFTB0900
      BBB = (R(NP2,J)*R(NP1)-R(NP1,J)*R(NP2))/DENOM                     AFTB0910
```

```
      N(NP3,J) = 1.2*P(J)+1.
 3011 CONTINUE                                                          AFTB0810
      CALL SHIFT (NP3,1)                                                AFTB0820
 3004 CONTINUE                                                          AFTB0830
      PPS(2) = PHI(I)-PHI(2)                                            AFTB0840
      PHI(2) = PHI(I)                                                   AFTB0850
      CON = (PPS*VEL*PS(I))**2                                          AFTB0860
      IF(INDEX .NE. 0)CON = CON*PS(I)**2                                AFTB0870
      PRAD = 0.42*CON                                                   AFTB0880
      SQUV = XPI*P AR                                                   AFTB0890
      IF(ITAPE.EQ.0)GO TO 3                                             AFTB0900
  4   CONTINUE                                                          AFTB0910
      READ(4) (PZ(I),I=1,237)                                           AFTB0920
      READ(4) (Z(I),I=1,1400)                                           AFTB0930
      READ(4) (ISPEC(I),I=1,144)
      READ(4) (ZDOY(I),I=1,108)
      READ(4) ARC                                                       AFTB0960
      READ(4) NRAY                                                      AFTB0970
      READ(4) IRCORR                                                    AFTB0980
  411 WRITE(6,412)IRCORR,NRAY                                           AFTB0990
  412 FORMAT(//11H RECORD NO=   ,2I5)                                   AFTB1000
      IF(NRAY.LT.IRCORR)GO TO 409                                       AFTB1010
      DO 3326 NREC2=1,NREC2                                             AFTB1020
 3326 READ(2)                                                           AFTB1030
      GO TO 4                                                           AFTB1040
    3 CONTINUE                                                          AFTB1050
      P(1) = PRAY                                                       AFTB1060
      Z(1) = ZRAY                                                       AFTB1070
      DO 2 IT=1,NPTS                                                    AFTB1080
      ELT(IT) = T(IT)*TEMP                                              AFTB1090
      CH(IT) = 0.5*XL*PRA*XN(IT)**2                                     AFTB1100
      CPP(IT) = 0.0                                                     AFTB1110
      XMB(IT) = 0.0                                                     AFTB1120
      CALL SMALLH(T(IT),EPS(IT,1),EPS(IT,2))                           AFTB1130
      DO 305 J=1,NSPEC                                                  AFTB1140
      CPP(IT) = CPP(IT)+P(J)*GAMMA(IT,J)                                AFTB1150
      CH(IT) = CH(IT)+H(J)*GAMMA(IT,J)                                  AFTB1160
  305 XMB(IT) = XMB(IT)+GAMMA(IT,J)                                     AFTB1170
      XMB(IT) = 1.0/XMB(IT)                                             AFTB1180
      IF(P(IT).EQ.0.)
     1GO TO (IT) = 0.0  (IT)*  T(IT)/(XL*PRA* XMB(IT))                  AFTB1190
      IF(T(IT).EQ.0.)
     1IT(IT) = P(IT)*XL*PRA*XN (IT)/P C(IT)                            AFTB1200
      CALL EQUIL (IT)                                                   AFTB1210
      IF(NSEL.EQ.0)CALL PERF (IT,1)                                     AFTB1220
      EL(IT) = EQ(IT)                                                   AFTB1230
```

```
      PL(IT) = P(IT)                                                    AFTB1240
      DTL(IT) = 1.0                                                     AFTB1250
      TL(IT) = T(IT)                                                    AFTB1260
      CALL SECANT(IT)                                                   AFTB1270
    2 CONTINUE                                                          AFTB1280
      RAT(1) = RTIO                                                     AFTB1290
    4 CONTINUE                                                          AFTB1300
      READ(5,1) TITLE                                                   AFTB1310
      READ(5,1) ITCOV,NFIT,MDELTY,MITER,MCHI,MMAX,ITAPE,NROTAT,NOBLST   AFTB1320
      READ(5,3) ZMAX,ZETAMX,CRITIT,CRST                                 AFTB1330
      READ(5,1)MODRAY,NRAOUT,MXOBS,MPRMS,MXCOM,NPRM4                    AFTB1340
    3 FORMAT(8F12.7)                                                    AFTB1350
      READ(5,1245)ION,(C(J),J=1,ION)                                    AFTB1360
1245  FORMAT(I5,6F12.0/(8F12.0))                                        AFTB1370
      CHI = 10.0**(-MCHI)                                               AFTB1380
      ERROR = CRIT                                                      AFTB1390
      IF(ITAPE.NE.0)GO TO 88                                            AFTB1400
      IF(R(1).GT.R(2))GO TO 87                                          AFTB1410
      DIST1 = SQRT((R(1)-R(2))**2+(Z(1)-Z(2))**2)                       AFTB1420
      DIST2 = SQRT((R(1)-R(3))**2+(Z(1)-Z(3))**2)                       AFTB1430
      IF(DIST1.GT.(0.25*DIST2))GO TO 88                                 AFTB1440
   87 CONTINUE                                                          AFTB1450
      DO 89 I=3,NPTS                                                    AFTB1460
   89 CALL SHIFT(I,I-1)                                                 AFTB1470
      NPTS = NPTS-1                                                     AFTB1480
   88 CONTINUE                                                          AFTB1490
      IF(NOBLST.EQ.0)GO TO 83                                           AFTB1500
                  CALL BLAYST(PRES,TEMP,DENS,VEL,RSH,WGHT,Z(1),R(1),    AFTB1510
     *  PHI(1),U(1),RHO(1),P(1),T(1),PSI(1),HINF,BIGAM,BD1,BD2,BDC,ZDY,AFTB1520
     *  YDY,DZ(3),DR(3),DIFUN(7),DDELTA(7),DPSI(3),XPI,SEAR,GSUM)       AFTB1530
   83 XMIN=1.0E-30                                                      AFTB1540
      YMIN=1.0E-02                                                      AFTB1550
      RETURN                                                            AFTB1560
      END                                                               AFTB1570
                                                                        AFTB1580
                                                                        AFTB1590
```

```
$IBFTC NS01    LIST,DECK                                                        AFTA0000
       SUBROUTINE FRMAIN                                                        AFTA0010
C      FINITE RATE MAIN PROGRAM                                                 AFTA0020
       COMMON/EXTR/DSUMG(35)                                                    AFTA0030
       COMMON/THERMO/ HCFNT(7,15),FCFNT(7,15),EPSCNT(7,2),HJ0(15),              AFTA0040
      1               H(15),F(15),CP(15),EPSEQ(2),NATOMS(15)                   AFTA0050
       COMMON / CONTRL/ NSPEC,NVIB,NR,NC,ITAPE,NXDIV,DELX,INDIM,JOIN,           AFTA0060
      1               OMEGA,NRCORD,DELMIN,NITER,ERROR,TESTM,MODEL      ,        AFTA0070
      2               TEMP,PRES,DENS,VEL,WGHT,XLMBDA,RSH,S1  , TTEST   ,       AFTA0080
      4               NRUN,NBSIT,IPRNT,NOUT,RMSTST,CSST,ZETAMX,ZMAX,           AFTA0090
      5              INBLR,INDWT,INAFT,NBDY,ZOUTP(48),NRAD,IUNIT,ISW6,         AFTA0100
      6              SKAP,SLAM,GAMIST,NPTS,NMAX,DEVMAX ,MDELAY, TWALL,         AFTA0110
      7                  ITBOW,RMS,DEV,RSHP,S1P,XLAM,XKAP,XCST,XZETAM,         AFTA0120
      8                  ERRNEW,TSTGAM,NNMAX,NISW6,ZMAXN,NOUTN    ,            AFTA0130
      9                MINVAR , MAXVAR ,  PSIED(31)                            AFTA0140
       COMMON/FREE/  XO,PINF,GAMINF(15),EPSINF(2),VAM,PAM,RHOAM,UAM,           AFTA0150
      1              GAMAM(15),EPSAM(2),TAM,BIGAM,HINF      , WTAM             AFTA0160
       COMMON/BODY/  ZBDY(6),YBDY(6),BDI(6),BD2(6),BDN(6,6),                   AFTA0170
      1              BD1N(6),BD2N(6),ZBDN(6,6),ZBDY1(6),YBDY1(6)               AFTA0180
       COMMON/REACT/ THETA(2),TAUA(2),TAUB(2),TAUC(2),TAUD(2),C(2),            AFTA0190
      1              NUP(35,15),NU(35,15),AK(35),BK(35),CK(35),DK(35),         AFTA0200
      2              KFIND(35),CZI(35),CWI(35),CDI(35),A(35,2),                AFTA0210
      3              ALPIJ(6,15)                                               AFTA0220
       COMMON/AFTBDY/Z(35),R(35),RHO(35),XMW(35),PSI(35),                      AFTA0230
      1   RHI(35),BETA(35),CP(35),CL(35),P(35),T(35),U(35),GAMMA(35,15),       AFTA0240
      2   EPS(35,2),CPP(35),CH(35),CMACH(35),FC(35),C(35),FL(35),GL(35),       AFTA0250
      3   DSTL(35),PL(35),ELT(35)                                              AFTA0260
       COMMON/AFTBDY/DZ(3),DR(3),RHI(2),DELTA(3),PSI(3),BPI(3),                AFTA0270
      1   BXI(3),BOMEGA(3),CII(3),CRR(3),CRS(3),SIGMA(3),HO,IR,IS,IQ,          AFTA0280
      2   IS1,IS2,IS3,IQ1,IR2,IR3,NRAY,CPI,ST,XKT,XKP,ZBAY,ZETA(2),DPDS,       AFTA0290
      3   ICN,B(8,9),DTDS,CLAR,GBUF,XPI,MAXRAY,ISW1,ISW4,NAOUT,                AFTA0300
      4   NMOD6,NREM6,NMOD4,NREM4,IDRAY,ITAPE,IDRUP,IODX,IFIT,NFSUT,           AFTA0310
      5   RHETAT,RHOBL(3,3),DCON(3),ECTR(3),IBR(3),PIBAR(3),PIBAT(3),          AFTA0320
      6   PIERR(3),BETAS(3),BMIS(3),PSIBAR(3),DELBAR(3),PBL(3,2),              AFTA0330
      7   G(15),NROTAT,GAMMAO,DERDRY(2),DGAMDY(15),DERSDS(4,2),                AFTA0340
      8   DGAMDS(4,15)                                                         AFTA0350
       DIMENSION W(36,1)                                                       AFTA0360
       EQUIVALENCE(Z,W)                                                        AFTA0370
       DO 77 I=1,1507
   77  Z(I)=0.0
       REWIND 4                                                                AFTA0380
   20  IR = 1                                                                  AFTA0390
       IS = 2                                                                  AFTA0400
       IQ = 3                                                                  AFTA0410
       IS1 = 4                                                                 AFTA0420
       IS2 = 5
```

```
      IS2 = 6                                                          AFTA0430
      IB1 = 7                                                          AFTA0440
      IB2 = 8                                                          AFTA0450
      IB3 = 9                                                          AFTA0460
      BOMEGA(2) = 1.234567
      CALL START                                                       AFTA0470
      OMEGA = INDIM                                                    AFTA0480
      IBODY=0                                                          AFTA0490
      IF(INBLR.EQ.0)IBODY=1                                            AFTA0500
      IF(ITAPE.EQ.1) GO TO 42                                          AFTA0510
      DO 25 I=1,NPTS                                                   AFTA0520
   25 NUM(I)=I                                                         AFTA0530
      NRAY=0                                                           AFTA0540
      NDROP = 0                                                        AFTA0550
      DO 30 I=1,NPTS                                                   AFTA0560
   30 CALL SHIF (NPTS+1-I,NPTS+IB3-I)                                  AFTA0570
      DO 31 I=2,IB3                                                    AFTA0580
   31 CALL SHIF (NPTS+IB3,I-1)                                         AFTA0590
      CALL SHIF  (NPTS+IB3-1,IS2)                                      AFTA0600
   40 DO 41 I=1,IB3                                                    AFTA0610
   41 NUM(I)=I-IB3                                                     AFTA0620
   42 CONTINUE                                                         AFTA0630
      IDROP = 0                                                        AFTA0640
      IF(NRCORD.EQ.0) GO TO 800                                        AFTA0650
      IF(MOD(NRAY,MODRAY).NE.NRAOUT) GO TO 70                          AFTA0660
  800 CALL OUTPUT (IB3,NPTS+IB3-1)                                     AFTA0670
      CALL STIF(NSTOP)                                                 AFTA0680
      IF(NSTOP.EQ.1) GO TO 700                                         AFTA0690
   70 NRAY=NRAY+1                                                      AFTA0700
      IF(NSTOP)800,50,800                                              AFTA0710
   50 CONTINUE                                                         AFTA0720
      CALL RAY                                                         AFTA0730
      DO 80 I=1,NPTS                                                   AFTA0740
   80 CALL SHIF (NPTS+IB3-I,NPTS+IB3+MDELAY+1-I)                       AFTA0750
  211 CONTINUE                                                         AFTA0760
      KDROP = 0                                                        AFTA0770
      IF(NTAPE-NROTAT)86,86,83
   83 IF(IBODY.EQ.0)GO TO 89                                           AFTA0790
      CALL BODYB(IB3+MDELAY+1)                                         AFTA0800
      GO TO 86                                                         AFTA0810
   89 CALL COUPLE(IB3+MDELAY+1)                                        AFTA0820
   86 CALL SHOCK (IB3 + MDELAY + NPTS)                                 AFTA0830
      IDROP=0                                                          AFTA0840
      DO 130 I=2,NPTS                                                  AFTA0850
      IK=IB3+MDELAY+I                                                  AFTA0860
      IT=IB3+I-1-IDROP                                                 AFTA0870
```

```
                                                                        AFTA0880
                                                                        AFTA0890
                                                                        AFTA0900
                                                                        AFTA0910
                                                                        AFTA0920
                                                                        AFTA0930
                                                                        AFTA0940
                                                                        AFTA0950
                                                                        AFTA0960
                                                                        AFTA0970
                                                                        AFTA0980
                                                                        AFTA0990
                                                                        AFTA1000
                                                                        AFTA1010
                                                                        AFTA1020
                                                                        AFTA1030
                                                                        AFTA1040
                                                                        AFTA1050
                                                                        AFTA1060
                                                                        AFTA1070
                                                                        AFTA1080
                                                                        AFTA1090
                                                                        AFTA1100
                                                                        AFTA1110
                                                                        AFTA1120
                                                                        AFTA1130
```

```
*LIST,NOSEQ
      SUBROUTINE RAY
      COMMON/THERAC/ HCENT(7,15),ECENT(7,15),EPSCNT(7,2),HJD(15),
     1               H(15),E(15),CP(15),EPSEQ(2),NATOMS(15)
      COMMON / CONTRL/ IPREC,NVIB,NR,NC,ITAPE,MXDIV,DELX,INCIM,ICIN,
     1               OMEGA,NRCORE,DELMIN,NITER,ERROR,TEST,MODEL ,
     2               TEMP,PRES,DENS,VEL,WGHT,XLAMDA,RMH,SI , ITEST ,
     4               ARC ,NOSIT,IPRNT,NOUT,RMSI  I,CSSI,ZETAMX,ZMAX,
     5               INCLR,INDXT,INAFT,NBODY,ZOUTP(4Y),NRAD,IUNIT,ISW6,
     6               SKIP,SLAM,GAMTST,NPTS,IMAX,DEVMAX ,MDELAY, IALL,
     7                  ITSOW,KMP,DEV,RSHP,SIP,XLAM,XSHP,XCST,XZETAM,
     8                  ERRMEW,TOTCAY,NTMAX,NISW6,ZMAXN,NCUTN  ,
     9                  MINVAR , MAXVAR , PSIED(21)
      COMMON/EREF/ XO,PIMP,GAMINF(15),EPSINF(2),VAM,PAM,RHOAM,UAM,
     1               GAMAM(15),EPSAM(2),TAM,SIGAM,HINF
      COMMON/ZODY/  ZODY(6),YBDY(6),BDI(6),BD2(6),BDS(6,2)
      COMMON/REACT/ THETA(2),TAUA(2),TAUB(2),TAUC(2),TAUD(2),Q(2),
     1               NUF(35,15),NU(35,15),AK(35),BK(35),CK(35),DK(35),
     2               KFIIND(35),CZI(35),CZI(35),CDI(35),A(35,2),
     3               ALPIJ(8,15)
      COMMON/AFTBDY/Z(35),R(35),NUM(35),RHO(35),XMM(35),PSI(35),
     1    PHI(35),BETA(35),CA(35),CB(35),P(35),T(35),U(35),GAMMA(35,15),
     2    RHS(35,2),CPP(35),CH(35),CMACH(35),PC(35),Q(35),FL(35),GL(35),
     3    RSTL(35),PLT(35),SLT(35)
      COMMON/AFTBDY/Z2(3),R2(3),THI(3),DELTA(3),PSI(3),PI(3),
     1    AXI(3),BOMEGA(3),C1S(3),C2S(3),C3S(3),SIGMA(3),HC,IR,IS,IG,
     2    IS1,IS2,IS3,IT1,IT2,IT3,NRAY,SHI,ST,XMD,XMP,ZRAY,ZETA(2),DPDS,
     3    IDN,R(8,4),DTDS,SCAR,SSJT,XPI,MAXRAY,IL 1,IL 4,NPOUT,
     4    NMOD6,ARCS,NX 4,NPE 4,MDIFRY,MTRL,IDRIP,ILOSS,MFIT,SHL T,
     5    PUETAT,RHOEL(3,3),DCONT(3),PCON(3),IEE(3),PIDER(3),PIMP(3),
     6    PIDER(3),DETAB(3),PHID(3),PCILAM(3),DELTAX(3),FLL(3,3),
     7    Q(15),NROTAT,GAMMAS,DEPSI(2), ERMS(15),DEPSDL(4,2),
     8    DGAMDS(4,15)
      COMMON/ARCLEN/ARC
      COMMON/BLOCK1/W(35,7)
      DIMENSION W(35,1)
      EQUIVALENCE (W(1,1),R(1))
      TANGT(X) = SIN(X)/COS(X)
      BOMEGA(3) = BPHI(3)+BETA(10)
      BOMEGA(2) = BOMEGA(2) +1.0
      IF(BOMEGA(1).EQ.1.234567)GO TO 405
      IF(NPUNCH.EQ.1.AND.BOMEGA(3).LT.1.5446184) GO TO 410
      GO TO 405
  410 NROTAT = IFIX(PHI(9)*57.2957795/2.0) + 1
      BOMEGA(1) = 1.234567
      NTAPE = 0
```

```
      BOMEGA(2) = 2.234567
 40[illegible] IF(NROTAT.EQ.0) GO TO 45
      IF(NPUNCH.EQ.1.AND.BOMEGA(3).LT.1.5446164) GO TO 415
      GO TO 420
 415  NTAPE = NTAPE+1
      IF(NTAPE.GT.NROTAT) GO TO 20
      GO TO 425
 420  NTAPE = NRAY
      IF(NRAY.GT.NROTAT) GO TO 20
 425  IF(BOMEGA(2).NE.2.234567) GO TO 55
 51   CALL [illegible]
      [illegible]T = [illegible]Z(3)
      [illegible]T = FZ
      ROT = NROTAT
      DTAN = (TANECT([illegible] (3))- FFZ)/ROT
      [redacted]DTAN = BPHI(3)/ROT
 [illegible] IK = [illegible]
      [illegible]Z(3) = Z(IK)
      [illegible]R(3) = R(IK)
      [illegible](1) = [illegible](IK)
      [illegible]Z(2) = [illegible]Z(3)
      [illegible](2) = [illegible](3)
      [illegible]Z(1) = Z(IK)
 [illegible] [illegible](1) = [illegible](3)
      IF(NPUNCH.EQ.1.AND.[redacted] GO TO 53
      [illegible]I(3) = TANE[illegible]T( [illegible]HI(3))-[illegible]
      [illegible]I(3) = [illegible]TAN( [illegible](3))
      GO TO 52
 53   BPHI(3)[redacted]DTAN
 52   [illegible](2) = [illegible](1)+([illegible](3)-[illegible](1))/2.0
      [illegible]                (6,1[illegible]) ([illegible](J), [illegible](J), [illegible](J),J=[illegible])
 10[illegible] [illegible]T(//2X,1P[illegible]2.[illegible])
 4[illegible] [illegible](1) = SIG[illegible](3)
      IF([redacted]ROTAT) GO TO 40
 [illegible] SIGMAT=[illegible](3)+[illegible]5*(SI[illegible](3)-SI[illegible](1))
      [redacted]
      IF([illegible].[illegible]0)[illegible] TO 5000
      [illegible]Z(3) = [illegible]Z3T
      [illegible]R(3) = [illegible]R3T
 5000 SIG[illegible] = (SIG[illegible](3)+SIGMAT)/2.0
      [illegible] = [illegible]RT(X[illegible]/1.4)
      [illegible]=([illegible] *[illegible] (SIGMAT))**2
      [illegible]
      [illegible]  -[illegible]*([illegible]2-1.) (7.*[illegible]))
      [illegible] /((7.*[illegible]2-1.)+(9[illegible].))
      IF([illegible]-1.) 21,21,[illegible]
```

```
      CALL BCHSE(21,6HRAY    )
   22 BETAT=ATAN (1./SQRT (XX**2-1.))
      PHIT=ATAN (1./(SIN (SIGMAT)/COS (SIGMAT)*(5.*CVG**2/(5.*(G**2-1.))
     *     -1.)))
      IF (NEWRAY.NE.1) CONST = BETAYX
      NEWRAY = 1
      N2 = IS2 + 2
      N1 = IB1 + NPTS
      BETAMX = 0.0
      PHIMAX = 0.0
      DISTMN = 100.0
      DO 1 N=N2,N1
      DIST = (Z(N)-Z(N-1))**2 + (R(N)-R(N-1))**2
      IF (DISTMN.GT.DIST) DISTMN=DIST
      IF (BETA(N).GT.BETAMX) BETAMX=BETA(N)
    1 IF (PHI(N).GT.PHIMAX) PHIMAX = PHI(N)
      DISTMN = SQRT(DISTMN)
      XX =(AMACH+1)/2.
      BETAMX = CONST* COS(PHIMAX-BPHI(1))/SIN(BETAMX)*DISTMN/XX
      DSHOCK=BETAMX*COS (BETAT+PHIT-BPHI(3))/COS (BPHI(3)-SIGMAK)
      ZSHOCK=Z(IS2)+DSHOCK*COS (SIGMAK)
      RSHOCK=R(IS2)+DSHOCK*SIN (SIGMAK)
      NUMBER=0
      ZRAY = BZ(3)+DSHOCK*COS(SIGMAK)
      IF(ISW6-1) 30,25,30
   25 DSIGAR=SIGMAR*57.2957795
      DSIGMA=SIGMAT*57.2957795
      WRITE (6,32)
   32 FORMAT(1H1)
      WRITE          (6   ,26)NUMBER,ZSHOCK,RSHOCK,DSHOCK,PHIT,XX**2,SIGMAR,
     *                        DSIGAR,DSIGMA,NU,SIGMAK,BETAMX,CONST,DISTMN,
     2                        XX,PHIMAX,BPHI(1),BETAMX
   26 FORMAT( 2H RAY ITER I3,1P8E15.7)
   30 CALL BLYER (ZRAY,FZ,FPZ)
      ZRAYP=ZRAY
      NUMBER = NUMBER +1
      ZRAY = ZSHOCK+FPZ*(RSHOCK-FZ)
      IF(ISW6.NE.1)GO TO 35
   31 WRITE          (6   ,26)NUMBER,ZRAY,ZRAYP,FZ,FPZ
   35 IF (ABS ((ZRAY-ZRAYP)/ZRAY)-CHI) 40,40,36
   36 ZRAY =.5*(ZRAY+ZRAYP)
      IF (NUMBER - 50) 30,38,37
   37 CALL DUMP(37,6HRAY    )
   40 RETURN
      END
```

```
$IBFTC NS14     LIST,DECK                                                       AFTK0000
       SUBROUTINE STIP(NSTOP)                                                  AFTK0010
       COMMON/THERMO/ HCENT(7,15),FCENT(7,15),EPSCNT(7,2),HOO(15),             AFTK0020
      1               H(15),F(15),CP(15),EPSEG(2),NATOMS(15)                   AFTK0030
       COMMON / CONTRL/ NSPEC,NVIS,NR,NC,ITAPE,NXDIV,DELX,INDIM,JOIN,          AFTK0040
      1               OMEGA,NRCORD,DELMIN,NITER,ERROR,TESTM,MODEL      ,       AFTK0050
      2               TEMP,PRES,DENS,VEL,WGHT,XLMBDA,RSH,GI  , TTEST   ,       AFTK0060
      4               NRUN,NBSIT,IPRNT,NOUT,RMSTST,CGST,ZETAMX,ZMAX,           AFTK0070
      5              INBLR,INDWT,INAFT,NBDY,ZOUTP(48),NRAD,IONIT,ISW6,         AFTK0080
      6              SKAP,SLAM,GAMTST,NPTS,NMAX,DEVMAX ,MDELAY, TWALL,         AFTK0090
      7                  ITBOW,RMS,DEV,RSHP,GIP,XLAM,XKAP,XCGT,XZETAM,         AFTK0100
      8                  ERRNEW,TSTGAM,NNMAX,NISW6,ZMAXN,NOUTN    ,            AFTK0110
      9                 MINVAR , MAXVAR ,  PSIED(31)                          AFTK0120
       COMMON/FREE/ X0,PINF,GAMINF(15),EPSINF(3),VAM,PAM,RHOAM,JAM,            AFTK0130
      1               CAMAM(15),EPSAM(2),TAM,GIGAM,HINF      , WTAM            AFTK0140
       COMMON/BODY/  ZBDY(6),YBDY(6),BD1(6),BD2(6),BDL(6,5)                    AFTK0150
      1             ,BDIN(6),BD2N(6),BDLN(5,6),ZBDYN(6),YBDYN(6)              AFTK0160
       COMMON/REACT/ THETA(2),TAUA(2),TAUB(2),TAUC(2),TAUD(2),O(2),            AFTK0170
      1               NUP(35,15),NU(35,15),AK(35),BK(35),CK(35),DK(35),        AFTK0180
      2               KFIIND(35),CZI(35),CWI(35),CDI(35),A(35,2),              AFTK0190
      3               ALPIJ(8,15)                                              AFTK0200
       COMMON/AFTBDY/Z(35),R(35),NUM(35),RHO(35),XMW(35),PSI(35),              AFTK0210
      1   PHI(35),BETA(35),CA(35),CB(35),P(35),T(35),U(35),GAMMA(35,15),       AFTK0220
      2   EPS(35,2),CPP(35),CH(35),CMACH(35),FC(35),G(35),FL(35),GL(35),       AFTK0230
      3   DSTL(35),PL(35),ELT(35)                                              AFTK0240
       COMMON/AFTBDY/BZ(3),BR(3),BPHI(3),BDELTA(3),BPSI(3),BPI(3),             AFTK0250
      1   BXI(3),BOMEGA(3),C1S(3),C2S(3),C3S(3),SIGMA(3),H0,IR,IS,IQ,          AFTK0260
      2   IS1,IS2,IS3,IB1,IB2,IB3,NRAY,CHI,SI,XKD,XKP,ZRAY,ZETA(2),DPDS,       AFTK0270
      3   ION,B(8,9),DTDS,BBAR,GSUM,XPI,MAXRAY,ISW1,ISW4,NRAOUT,               AFTK0280
      4   NMOD6,NREM6,NMOD4,NREM4,MODRAY,NTAPE,IDROP,IBODY,NPIT,RMSOUT,        AFTK0290
      5   RGETAT,RHOBL(3,3),BCON(3),PCON(3),IBB(3),PIBAR(3),PIHAT(3),          AFTK0300
      6   PIERR(3),BETAS(3),PHIS(3),PSIBAR(3),DELBAR(3),PBL(3,3),              AFTK0310
      7   Q(15),NROTAT,GAMMA0,DEPSDY(2),DGAMDY(15),DEPSDB(4,2),                AFTK0320
      8   DGAMDS(4,15)                                                         AFTK0330
       COMMON/ARCLEN/ARC                                                       AFTK0340
       COMMON/CAL3D/NPUNCH                                                     AFTK0350
       DIMENSION W(35,1)                                                       AFTK0360
       EQUIVALENCE(Z,W)                                                        AFTK0370
       DIMENSION ISPEC(1)                                                      AFTK0380
       EQUIVALENCE(NEQUIL,NGAS),(ISPEC,NSPEC)                                  AFTK0390
       DIMENSION XEPSN(2),XEPS(21,2)                                           AFTK0400
       IF(NEARUN.NE.1)NRCORD=0                                                 AFTK0410
       NEARUN = 1                                                              AFTK0420
       NSTOP = 0                                                               AFTK0430
       ISW4 = 0                                                                AFTK0440
       ISW6 = 0                                                                AFTK0450
```

```
      IF(MOD(NRAY,NMOD4).EQ.NREM4) ISW4=1                                AFTK0440
      IF(MOD(NRAY,NMOD6).EQ.NREM6) ISW6          =1                      AFTK0450
      IF(NRCORD.EQ.0) GO TO 409                                          AFTK0460
      IF(MAXRAY.NE.0.AND.NRAY.GE.MAXRAY)NSTOP=1                          AFTK0470
      IF(ZMAX.NE.0.0.AND.BZ(3).GE.ZMAX)NSTOP=1                           AFTK0480
      IF(NSTOP.EQ.1) GO TO 409                                           AFTK0490
      IF(MOD(NRAY,MODRAY).NE.NRAOUT) GO TO 50                            AFTK0500
  409 CONTINUE                                                           AFTK0510
      WRITE(4) (BZ(I),I=1,297)                                           AFTK0520
      WRITE(4) (Z(I),I=1,1400)                                           AFTK0530
      WRITE(4) (ISPEC(I),I=1,144)                                        AFTK0540
      WRITE(4)  (ZBDY(I),I=1,108)                                        AFTK0550
      WRITE(4) ARC                                                       AFTK0560
      WRITE(4) NRAY                                                      AFTK0570
  410 WRITE (4) NRCORD                                                   AFTK0580
  411 WRITE(6,412) NRCORD                                                AFTK0590
  412 FORMAT(//11H RECORD NO  ,I5)                                       AFTK0600
      NRCORD = NRCORD+1                                                  AFTK0610
      IF(NPUNCH.EQ.1.AND.BOMEGA(3).LT.1.5446164) GO TO 418
      RETURN
  418 IF(NTAPE.GE.NROTAT) GO TO 419
      RETURN
  419 CONTINUE
      NSTOP = 1
  420 FORMAT(5I5,5X,I5,10X,I5)                                           AFTK0630
  421 FORMAT(5E15.7)                                                     AFTK0640
  422 FORMAT(15X,2E15.7)                                                 AFTK0650
      WRITE(9,420)ITAPE,NPLIN                                            AFTK0660
      NDELAY = 3                                                         AFTK0670
      WRITE(9,420) NSPEC,NR,NC,NVIB,MODEL,NPTS,NDELAY                    AFTK0680
      DSIG1= 57.2957795*SIGMA(1)
      DSIG3= 57.2957795*SIGMA(3)
      WRITE(9,421) DSIG1,DSIG3
      RO = 8.313405E+07                                                  AFTK0700
      WRITE(9,421) DENS,VEL,RO,TEMP,WGHT,RSH,HINF,BZ(3)                  AFTK0710
      DO 425 K =1,2                                                      AFTK0720
      XEPSN(K) = EPSINF(K)
      IF(NVIB.EQ.0)GO TO 114
      GO TO 112
  112 IF(NVIB.EQ.1.AND.K.EQ.2)GO TO 114
      GO TO 425
  114 EPSINF(K) = 0.0
  425 CONTINUE                                                           AFTK0750
      WRITE(9,421) BIGAM,EPSINF(1),EPSINF(2)                             AFTK0760
      IF(NVIB.EQ.2)GO TO 436
      DO 435 K=1,2
```

```
435 EPSINF(K) = XEPSN(K)
436 WRITE(9,422) CHI                                                    AFTK0770
    WRITE(9,421) (GAMAM(J),J=1,NSPEC)                                  AFTK0780
    IPTS8 = NPTS + 8                                                   AFTK0790
    DO 430 I=9,IPTS8                                                   AFTK0800
    WRITE(9,421) PHI(I),R(I),P(I),U(I),T(I),RHO(I),XMW(I)              AFTK0810
    DO 426 K=1,2                                                       AFTK0820
    XEPS(I,K) = EPS(I,K)
    IF(NVIB.EQ.0)GO TO 503
    GO TO 502
502 IF(NVIB.EQ.1.AND.K.EQ.2)GO TO 503
    GO TO 426
503 EPS(I,K) = 0.0
426 CONTINUE                                                           AFTK0850
    WRITE(9,421) EPS(I,1),EPS(I,2)                                     AFTK0860
    IF(NVIB.EQ.2)GO TO 504
    DO 505 K=1,2
505 EPS(I,K) = XEPS(I,K)
504 WRITE(9,421) (GAMMA(I,J),J=1,NSPEC)                                AFTK0870
430 CONTINUE                                                           AFTK0880
415 CONTINUE                                                           AFTK0890
 50 RETURN                                                             AFTK0900
    END                                                                AFTK0910
```

The axially symmetric program of Reference 60 ordinarily uses a mesh composed of streamlines and rays normal to the body surface. In order to prepare input data for the present program it is desirable to rotate these rays about a fixed point on the body surface until they are normal to the wind axis. In doing so, however, it is essential that the resulting ray be located so that the backward characteristics from the first plane computed by the 3-D program intersect the initial data plane. This condition is equivalent to requiring that the angle

$$\Omega = \beta + \phi < 90°$$

The indicated program changes will rotate the rays through successive 1-degree increments as soon as it is safe to do so ($\Omega < 88.5°$) and will then punch the necessary data in the proper format. FORTRAN unit 9 is used in the program as written above, but this may, of course, be altered to suit the requirements of the local computing system.

C.2 INPUT PROCEDURE

The input to the axially symmetric program is set up as described in Reference 60, VOL. II. If the ray rotation and punching are desired, the user must punch a 1 in column 20 of "Card Y" (Ref 60, p. 14 Vol. II p. 14).

DISTRIBUTION LIST

Activities Located at Wright-Patterson Air Force Base, Ohio

APR	Hypersonic Research Laboratory	1 cy
SES	Deputy for Studies & Analyses	1 cy
SESSV	Vehicle Design Division	1 cy
FDD	Vehicle Dynamics Division	1 cy
FDC	Flight Control Division	1 cy
FDT	Structures Division	1 cy
SEN	Deputy for Systems Engineering	1 cy
SESS	Directorate of Analysis	1 cy
FDP	(STINFO)	1 cy
ASZ	Deputy for Systems Management	1 cy
ARIR	ARL - Technical Documents Report	1 cy
FDE	RTD Library	1 cy
TDF	Aerospace Weapon Systems Division	2 cys

Flight Dynamics Laboratory

FDX	Chief Scientist	1 cy
FDM	Flight Mechanics Division	1 cy
FDME	Electrogasdynamics Test Branch	1 cy
FDMG	Gasdynamics Branch	10 cys
FDMS	Aerospace Vehicle Branch	1 cy
FDMM	Aeromechanics Branch	1 cy
AFIT	Air Force Institute of Technology	1 cy
TDBTL	Foreign Technology Division (Technical Library)	1 cy

Unclassified

Security Classification

DOCUMENT CONTROL DATA - R & D

Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY (Corporate author)
Cornell Aeronautical Laboratory, Inc.

2a. REPORT SECURITY CLASSIFICATION
Unclassified

2b. GROUP

3. REPORT TITLE
The Method of Characteristics for Three-Dimensional Steady Real Gas Flows

4. DESCRIPTIVE NOTES (Type of report and inclusive dates)
Final Report 1 April 1963 - November 1966

5. AUTHOR(S) (First name, middle initial, last name)
Strom, Charles R.

6. REPORT DATE	7a. TOTAL NO. OF PAGES	7b. NO. OF REFS
March 1967		

8a. CONTRACT OR GRANT NO	9a. ORIGINATOR'S REPORT NUMBER(S)
AF 33(657)-10423	AM-1800-Y-4
b. PROJECT NO	
c.	9b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report)
d.	AFFDL-TR-67-57

10. DISTRIBUTION STATEMENT
Distribution of this document is unlimited.

11. SUPPLEMENTARY NOTES	12. SPONSORING MILITARY ACTIVITY
	Air Force Flight Dynamics Laboratory Research & Technology Division Wright-Patterson AFB, Ohio

13. ABSTRACT

The method of characteristics is formulated for the computation of the supersonic flow of an inviscid, reacting gas over a smooth three-dimensional body. Various methods of constructing networks of bicharacteristic lines are examined from the point of view of numerical stability and accuracy. A new method of forming the network, which consists of projecting forward along streamlines from data points on specified data planes, is found to be most easily adopted to the particular requirements of nonequilibrium chemistry.

The general method was coded for the IBM 7090 computer and the program demonstrated for the case of an ideal gas. Calculations were made for the case of an ideal gas. Calculations were made for the flow about a spherical-tip 15° half-angle cone at 10° angle of attack and a generalized elliptical body at zero incidence. Since the program yields the pressure distribution along specified streamlines, it is straightforward, in principle, to link it to a finite-rate chemistry stream tube program to treat three-dimensional, nonequilibrium flows.

DD FORM 1 NOV 65 1473

Unclassified

Security Classification

Unclassified

Security Classification

14. KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
Method of Characteristics, Three-D ensional Flow, Numerical Procedures for Three-Dimensional Flows, Nonequilibrium Chemistry, Simplical and Non-Simplical Networks.						

INSTRUCTIONS

1. ORIGINATING ACTIVITY. Enter the name and address of the contractor, subcontractor, grantee, Department of Defense activity or other organization *(corporate author)* issuing the report.

2a. REPORT SECURITY CLASSIFICATION: Enter the overall security classification of the report. Indicate whether "Restricted Data" is included. Marking is to be in accordance with appropriate security regulations.

2b. GROUP: Automatic downgrading is specified in DoD Directive 5200.10 and Armed Forces Industrial Manual. Enter the group number. Also, when applicable, show that optional markings have been used for Group 3 and Group 4 as authorized.

3. REPORT TITLE: Enter the complete report title in all capital letters. Titles in all cases should be unclassified. If a meaningful title cannot be selected without classification, show title classification in all capitals in parenthesis immediately following the title.

4. DESCRIPTIVE NOTES: If appropriate, enter the type of report, e.g., interim, progress, summary, annual, or final. Give the inclusive dates when a specific reporting period is covered.

5. AUTHOR(S): Enter the name(s) of author(s) as shown on or in the report. Enter last name, first name, middle initial. If military, show rank and branch of service. The name of the principal author is an absolute minimum requirement.

6. REPORT DATE: Enter the date of the report as day, month, year, or month, year. If more than one date appears on the report, use date of publication.

7a. TOTAL NUMBER OF PAGES: The total page count should follow normal pagination procedures, i.e., enter the number of pages containing information.

7b. NUMBER OF REFERENCES: Enter the total number of references cited in the report.

8a. CONTRACT OR GRANT NUMBER: If appropriate, enter the applicable number of the contract or grant under which the report was written.

8b, 8c, & 8d. PROJECT NUMBER: Enter the appropriate military department identification, such as project number, subproject number, system numbers, task number, etc.

9a. ORIGINATOR'S REPORT NUMBER(S): Enter the official report number by which the document will be identified and controlled by the originating activity. This number must be unique to this report.

9b. OTHER REPORT NUMBER(S): If the report has been assigned any other report numbers *(either by the originator or by the sponsor)*, also enter this number(s).

10. AVAILABILITY/LIMITATION NOTICES: Enter any limitations on further dissemination of the report, other than those imposed by security classification, using standard statements such as:

(1) "Qualified requesters may obtain copies of this report from DDC."

(2) "Foreign announcement and dissemination of this report by DDC is not authorized."

(3) "U. S. Government agencies may obtain copies of this report directly from DDC. Other qualified DDC users shall request through _____."

(4) "U. S. military agencies may obtain copies of this report directly from DDC. Other qualified users shall request through _____."

(5) "All distribution of this report is controlled. Qualified DDC users shall request through _____."

If the report has been furnished to the Office of Technical Services, Department of Commerce, for sale to the public, indicate this fact and enter the price, if known.

11. SUPPLEMENTARY NOTES: Use for additional explanatory notes.

12. SPONSORING MILITARY ACTIVITY: Enter the name of the departmental project office or laboratory sponsoring *(paying for)* the research and development. Include address.

13. ABSTRACT: Enter an abstract giving a brief and factual summary of the document indicative of the report, even though it may also appear elsewhere in the body of the technical report. If additional space is required, a continuation sheet shall be attached.

It is highly desirable that the abstract of classified reports be unclassified. Each paragraph of the abstract shall end with an indication of the military security classification of the information in the paragraph, represented as *(TS)*, *(S)*, *(C)*, or *(U)*.

There is no limitation on the length of the abstract. However, the suggested length is from 150 to 225 words.

14. KEY WORDS: Key words are technically meaningful terms or short phrases that characterize a report and may be used as index entries for cataloging the report. Key words must be selected so that no security classification is required. Identifiers, such as equipment model designation, trade name, military project code name, geographic location, may be used as key words but will be followed by an indication of technical context. The assignment of links, roles, and weights is optional.

DD FORM 1 JAN 64 1473 (BACK)

Unclassified

Security Classification